Sudan University of Science and Technology

College of Graduate Studies

Developing a Model for Optimizing Video Streaming Quality of Experience in Software-defined Networks

تطوير نموذج لتحسين جودة التجربة لتدفق الفيديو في الشبكات المعرفة برمجي

A Thesis Submitted in Fulfillment of the Requirements for Ph.D. Degree in Computer Science

By

Majda Omer Elbasheer Ali

Supervised by

Prof. Jaime Lloret Mauri

August 2022

DECLARATION by STUDENT

I hereby certify that this material is entirely my own investigation and work, which was completed while registered as a candidate for the degree of Doctor of Philosophy. I also declare that it has not been previously or concurrently submitted as a whole for any other degrees at the Sudan University of Science and Technology or other institutions.

Student's name: **Majda Omer Elbasheer Ali**

Signature

Date August; 16,2022

DECLARATION by MAIN SUPERVISOR

This is to certify that the thesis entitled ***"Development of a Model for Optimizing Video Streaming Quality of Experience in Software-defined Networks***" presented by ***Majda Omer Elbasheer Ali*** is an original work that she has done under my guidance and supervision to fulfill the requirements for the Ph.D. degree in computer science at Sudan University of Science and Technology. Additionally, I certify that the thesis was prepared and completed in accordance with the guidelines of the College of Graduate Studies.

Supervisor's name: **Prof. Jaime Lloret Mauri**

Supervisor's signature:

Institut d'Investigació per a la gestió integrada de zones costaneres
IGIC
UNIVERSITAT POLITÈCNICA DE VALÈNCIA
CAMPUS DE GANDIA

Date August; 16,2022

DEDICATION

To my dear parents, to my wonderful husband, and to my lovely daughters. It would not have been possible for me to complete my thesis without their support.

ACKNOWLEDGMENT

I would like to thank all the people who made my dissertation possible. I am indebted to them for their support and cooperation; however, given the limits of this document, I can only mention a few of them.

First of all, I would like to express my sincere gratitude to my supervisor, Prof. Jaime Lloret Mauri from the University of Politecnica de Valencia, for his continuous support of my Ph.D. study and research, as well as for his patience and immense knowledge. His guidance has helped me throughout the time of research and writing of this thesis.

I am grateful to my husband Mozamil Hassan for his encouragement, assistance, support, and faith in me to take on this huge task. I would also like to thank my parents for being a source of inspiration to me, and to my brother and sisters for their love and support. It is also my pleasure to thank my sister-in-law Safaa for taking care of my little daughters during the completion of this thesis.

Lastly, a special thanks go out to my best friend Dalia Mahmoud Elsir for her constructive and insightful comments, and for all her help. My thanks and blessings to Omaima FathElrahman, as well as anyone else who contributed or supported me during the journey of this work.

ABSTRACT

In recent years, video streaming has become widely widespread, especially with the growth of users, mobile devices, and the availability and diversity of multimedia applications. Real-time video communications require quality of experience (QoE) awareness to provide satisfactory service to customers. Video streaming also requires a certain level of performance and a stable network to accommodate the quality of service (QoS) requirements of video users and applications, since QoS depends on network performance, this directly affects QoE.

The emergence of software-defined networking (SDN) could eliminate current network limitations. Additionally, SDN's flexible programmability and global view capabilities could facilitate automated QoS control and management.

The thesis proposed a video streaming adaptive QoS-based routing and resource reservation (VQoSRR) model, which gives SDN networks the ability to meet video demands and enhance user experience compared to the best-effort networks. In order to implement QoS-based routing (QBR), algorithms were developed for calculating routing, installing routing paths in the forwarding devices, and shifting traffic to an alternative path when QoE is violated. As well, queuing mechanisms were used to allocate resources based on the QoE requirements of video streaming. The traffic was differentiated based on video resolution QoE parameters to Standard Definition (SD) and High Definition (HD).

The Thesis results showed that resource reservation mechanisms combined with QoS-based routing provided effective control over routes and resources. Moreover, the results demonstrated that the proposed method obtained better viewing quality and increased the overall throughput of the network.

المستخلص

في السنوات الأخيرة، انتشر دفق الفيديو على نطاق واسع، لا سيما مع نمو المستخدمين و الأجهزة المحمولة وتوافر وتنوع التطبيقات الوسائط المتعددة . تتطلب اتصالات الفيديو في الوقت الفعلي وعيًا بجودة الخبرة (QoE) لتقديم خدمة مرضية للعملاء. كما يتطلب دفق الفيديو أيضًا مستوى معينًا من الأداء وشبكة مستقرة لاستيعاب متطلبات جودة الخدمة (QoS) لمستخدمي وتطبيقات الفيديو ونظرًا لأن جودة الخدمة تعتمد على أداء الشبكة، فإن هذا يؤثر بشكل مباشر على جودة الخبرة (QoE). لكن، الاعتماد على البنية التحتية للشبكة التقليدية وخدمة الإنترنت الحالية التي تقدم الخدمات بأفضل جهد لا يضمن جودة الخدمة.

أدي ظهور الشبكات المعرفة بالبرمجيات (SDN) إلى القضاء على قيود الشبكة الحالية. الـ SDN هي نهج يهدف إلى فصل طبقتي التحكم والبيانات (أو طبقة إعادة التوجيه) للشبكة لإدارة وتحسين أفضل. بالإضافة إلى ذلك، إن مرونة البرمجة في الـ SDN وإمكانية الرؤية الشاملة لبنية الشبكة يمكن أن تسهل التحكم الآلي في جودة الخدمة وإدارتها.

في هذه الأطروحة ، تم اقتراح نموذج توجيه وحجز الموارد (VQoSRR) المستند إلى جودة الخدمة (QoS)، والذي يمنح شبكات SDN القدرة على تلبية متطلبات الفيديو وتحسين تجربة المستخدم مقارنة بالشبكات ذات الجهد الأفضل. من أجل تنفيذ التوجيه المستند إلى جودة الخدمة (QBR)، تم تطوير خوارزميات لحساب التوجيه ، وتثبيت مسارات التوجيه في أجهزة إعادة التوجيه ، وتحويل حركة المرور إلى مسار بديل عند انتهاك قيود ال (QoE). كذلك، تم استخدام آليات قائمة الانتظار لتخصيص الموارد بناءً على متطلبات الخبرة لتدفق الفيديو. وايضا ميزنا حركة المرور على أساس معاييردقة الفيديو الخاصة بال QoE إلى دقة قياسية ((Standard Definition (SD)) و ودقة عالية ((High Definition (HD)).

أظهرت نتائج البحث أن آليات حجز الموارد جنبًا إلى جنب مع التوجيه المستند إلى جودة الخدمة قد وفرت تحكمًا فعالًا في المسارات والموارد. علاوة على ذلك، أظهرت هذة النتائج أن الطريقة المقترحة حصلت على جودة مشاهدة أفضل وزادت من الإنتاجية الإجمالية للشبكة.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

TERM	ABBREVIATION
APIs	Application Programming Interfaces
AVC	Advanced Video Coding
Bps	bits per second
CBR	Constant Bit Rate
DR-RA	Dynamic Traffic Re-Routing Algorithm
DSCP	Differentiated Services Code Point
DVD	Digital Video Disc
FDM	Floodlight Default Mode
FPS	Frame Rates
FR	Full reference
HD	High-Definition
HEVC	High-Efficiency Video Coding
HVS	Human Visual System
IPDV	IP Packet Delay Variation
IPLR	IP packet Loss Ratio
IPTD	IP Packet Transfer Delay
IPTV	Internet Protocol Television
ITU-T	International Telecommunication Union -Telecommunication
IUFP	Installing or Updating Flow Path
MOS	Mean Opinion Score
MPEG	Moving Picture Experts Group
MPLS	Multi-Protocol Label Switching
MPQM	Moving Picture Quality Metric
MSE	Mean Squared Error
MSE	Mean Square Error
MSU	Video Quality Measurement tool
ND	Not Defined
NR	No Reference
OVS	Open vSwitch
PLR	Packet Loss Rate
PSNR	Peak Signal-to-Noise Ratio
PVQM	Perceptual Video Quality Measure
QBR	QoS-based Routing
QoE	Quality of Experience
QoS	Quality of Service
QP	Quantization Parameter
RR	Reduced reference
RTCP	Real-Time Control Protocol
RTP	Real-Time Transport Protocol
RTSP	Real-Time Streaming Protocol
SD	Standard-Definition

SDN	Software-Defined Networking
SLAs	Service Level Agreements
SMM	Single Mixed Metrics
SSIM	Structural Similarity Index Metric
SVC	Scalable Video Coding
TCP	Transmission Control Protocol
TwoLLWPs	Two Lowest Loss Widest Paths
UDP	User Datagram Protocol
UHD	Ultra-High-Definition
VBR	Variable Bit Rate
VCEG	Video Coding Experts Group
VLC	VideoLAN Client
VMAF	Video Multimethod Assessment Fusion
VQM	Video Quality Model
VQM	Video QoS Mode
VQoSRR	Video Streaming Adaptive QoS Routing with Resource Reservation

LIST OF PUBLICATIONS

1. Elbasheer, M.O., Aldegheishem, A., Lloret, J. and Alrajeh, N., 2021. A QoS-Based routing algorithm over software-define networks. Journal of Network and Computer Applications, 194, p.103215.
2. Elbasheer, M.O., Aldegheishem, A., Alrajeh, N. and Lloret, J., 2022. Video Streaming Adaptive QoS Routing with Resource Reservation (VQoSRR) Model for SDN Networks. Electronics, 11(8), p.1252.

CHAPTER I

INTRODUCTION

1.1 Background and Motivation

Recently, there is a continuous growth of video consumption by internet users accompanied by a development in the technology of real-time video sessions such as IPTV (Internet Protocol television) and Internet video services. In 2021, video traffic make up 80% of global Internet traffic, according to the Cisco Global Forecast Highlights report (Cisco, 2016). Another ongoing trend that feeds this growth is the increasing number of Smartphone devices, social media users, the diversity of video applications, and the advancement of network technologies, such as Wi-Fi and 5G connections.

In Addition, Internet video streams with higher resolution definitions have become more popular recently. The same report estimates that the Standard-Definition (SD) video traffic decreased to 24.5% in 2021. In contrast, High-Definition (HD) video traffic raised to 56.3%, and Ultra-High-Definition (UHD) or 4K was 19.2% of Internet video traffic in 2021. Figure 1.1 shows the evolution of video consumption and adoption between 2016 and 2021 (Cisco, 2016). Streaming video demands guaranteed performance, and these types of applications are known as real-time (strict latency) and rate-critical (specific data rate) applications. A good Quality of Service (QoS) is essential to stream video over the network and enhance the quality of experience (QoE). Besides, displaying videos at a higher definition will maximize user-perceived quality, since the video resolution is one of the video content characteristics that indicate the level of detail in a video frame.

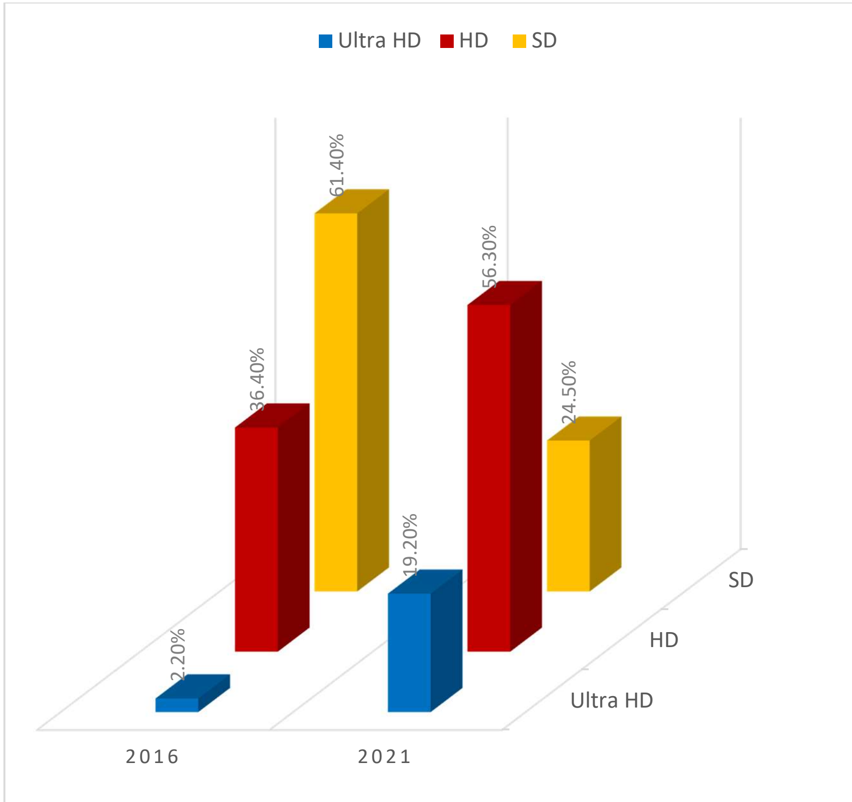

Figure 1.1: The forecast consumption and adoption from 2016 to 2021 according to different video resolutions (Cisco, 2016).

This increasing demand for high-quality online video requires network operators and media service providers to adopt new strategies and technologies. In this regard, QoS-based routing (QBR) was introduced to enable the routing layer to enhance traffic performance and overall QoS. The basic idea is to determine network paths according to various metrics to supply acceptable QoS for significant application flows, using some knowledge of resource availability in the network, in addition, to monitoring and adapting to QoS parameters variations at network links (Costa and Duarte, 1999). The QBR seeks to provide performance guarantees by mapping the multiple QoS requirements into routing metrics, such as bandwidth, delay, packet loss, etc. Also, it could exploit the best cost paths and non-best cost but acceptable paths (called feasible routes) (Crawley et al., 1998).

The various metrics used in path calculations could be handled as Single Mixed Metrics (SMM) or as multiple individual metrics (Wang and Crowcroft, 1996) (Costa et al., 2001). The first one computes and joins different QoS constraints in a single mathematical function. While the second applies many distinct metrics until it finds a feasible path that meets all quality restrictions. In this method, metrics are composed by using three different rules, additive (e.g.: delay), multiplicative (e.g.: packet loss), and concave (e.g.: bandwidth).

Furthermore, in traditional routing, source-based and hop-by-hop routing algorithms are proposed to achieve QoS routing. Source routing is based on the idea that the flow source is the one who calculates routes on-demand depending on the type of application, and it needs to have information about the entire network that is necessary to generate the forwarding paths. On the other hand, hop-by-hop algorithms make routing decisions by using the information available at each router. It allows to reduce the setup delays and distribute the overhead, still, the routers might not be able to avoid the routing loops.

In recent years, the emergence of Software-Defined Networking (SDN) architecture has allowed for innovative approaches to networking. SDN decouples the control plane and data plane, which has contributed to solving the problem of managing and controlling networks. In addition, it provides a global network view, where controllers can obtain complete topology information and statistics by using the OpenFlow protocols. We are therefore motivated to improve QoS architectures by leveraging the following SDN features:

- The logically centralized controller, besides maintaining state information for the flow path could facilitate writing quality-based routing algorithms and offer end-to-end QoS per flow. Also, the global view of resource availability enables traffic flows to be rerouted dynamically to ensure efficient resource utilization.

- OpenFlow forwarding device tables are constructed on a per-flow basis, so they are suitable for QoS routing because they are built according to traffic characteristics.
- Applying Service Level Agreements (SLAs) and QoS management policies easily by the administrator, that can be changed dynamically at a higher abstraction level, without the need of setting them at each forwarding device.

1.2 Problem Statement

Due to the special characteristics of video streaming and its applications, the networks must be managed to provide a convenient and guaranteed level of QoS and QoE. However, the QoS model in traditional networking faced unresolved issues such as per-hop decisions, limited network global view, and difficulty in providing different QoS abilities for various application flows. Besides, some of the existing IP short path routings algorithms cannot verify the link QoS required to adapt to the requested requirements of QoS for the flow; since it is unaware of the available QoS over the path (Crawley et al., 1998). In contrast, IP link-state protocols provide QBR by flooding updates to exchanging routing information and reflecting an up-to-date view of the network to calculate new routes. However, the frequent flooding process can impose significant communication overhead on forwarding devices, and repeatedly changing the routing paths can increase the delay experienced by the end-users (Crawley et al., 1998; Masip-Bruin et al., 2006). In addition to the short path routing problems above, the Internet best-effort service created a lack of deployment of QoS on the Internet architecture.

On the other hand, the QBR concept aims to provide routing solutions that could achieve multiple user's QoS requirements. However, the methods used in its routing decisions as single mixed metrics and multiple individual metrics; have issues associated with them. Firstly, the single-mixed metrics function cannot be sufficient for QoS routing because it is uncertain that each QoS requirement is respected. Also, due to the different composition rules of its parameters, it is difficult to define the mathematical composition rule for this method. Secondly, finding a path that meets multiple metrics of delay, cost, packet loss probability, and jitter is an NP-complete problem, according to Wang and Crowcroft (1996).

Finally, since the video is considered resource-intensive and consumes a lot of network bandwidth, especially videos with high resolution and bit rate. Thus, any planned model for video streaming must provide resource reservation capabilities. Further, increasing the video resolution makes any problems that occur during the delivery of video more apparent, causing different levels of degradation for the user QoE. As a result, it requires a good level of network quality of service (Pokhrel, 2015); because QoS parameters, such as jitter, delay, lost packets, etc., influence video QoE (Lloret et al., 2011).

1.3 The Research Hypotheses and Objectives

To solve the above problem, there are hypotheses to be tested in this thesis, namely that it is possible to propose an adaptive QoS-based routing and resource reservation model for video streaming over software-defined networks by developing methods for flow path selection and resource reservation based on video QoE requirements, thereby optimizing QoS, QoE, and overall resource utilization. In order to direct the research to test these hypotheses, the following detailed objectives have been set:

1. Study QoS techniques, algorithms, and network protocols that have been used to transfer video streaming. This has been accomplished by conducting a comprehensive analysis of existing proposals for video QoS in traditional networks and SDN networks.

2. Determine the performance parameters and metrics influencing QoS/QoE of video streaming.
3. Define selection criteria for QoS-based routes depending on the QoS requirements of video streaming flows.
4. Develop a management system to monitor and collect network performance information.
5. Design algorithms for calculating routing, installing routing paths in the forwarding devices, and shifting traffic to an alternative path when QoE is violated.
6. Propose an approach to provide effective bandwidth allocation for QoS flows.
7. Evaluate the performance of the proposed framework.

1.4 Research Methodology

This thesis adopts the following methodology, which contains the following phases, to achieve the predetermined objectives.

1.4.1 Proposed Model Design and Development

The model focused on designing and developing new applications, modifying the controller, and applying the proposed methods, criteria, and policies to the SDN control and application plane to enable video streaming over SDN. One of the tasks of the proposed model was to use the video QoS metrics to define criteria for measuring the performance of videos over the network.

In addition, a set of algorithms have been designed to provide a QoS-based routing strategy that supports video QoS by providing primary routes and alternative paths that satisfy multiple individual QoE metrics. The proposed algorithms determine feasible paths for each video flow based on the defined performance criteria and the current state of the network by taking advantage of the SDN controller's global view of the network. Moreover, a method for requesting and reserving network resources for network flows according to their importance has been incorporated to avoid video quality degradation. The queuing mechanisms and Differentiated Services Code Point (DSCP) were exploited to distinguish the network video traffic based on video resolution QoE parameters: Standard Definition and High Definition.

The proposed model was developed using Java and Python programming languages. The network and control modules were implemented in the Floodlight controller, an open-source, multi-threaded Java-based OpenFlow controller.

1.4.2 Evaluation and Validation Metrics

To investigate the performance of the proposed module at this phase, a network emulator tool, Mininet, was selected. Mininet simulates SDN and OpenFlow networks using Linux networking software, such as switches, controllers, and network performance parameters. A video client-server application was also implemented to exchange video streaming sessions.

Finally, to measure QoE the delivered video on each client was recorded and compared with the original video sequence. The simulation results were analyzed using two methods to evaluate the video quality of experience: subjective and objective measurements. Subjective metrics are performed by asking human subjects to rate the video they watched. On the other hand, objective evaluation of video QoE is based on objectively measured parameters of the network and media.

1.4.3 Publication of the research results and writing Thesis

In the final phase of the methodology, the research results were contributed to the research community, then the dissertation was written.

1.5 Contributions of the Thesis

The key contributions of this study are summarized below:

1. A management system was designed to collect and monitor performance data.
2. QBR algorithms were developed for path selection, rerouting traffic to an alternative path, and installing routing paths based on video streaming QoE requirements.
3. A higher-level reservation control strategy was defined to enable administrating of allocating bandwidth for the different flows according to their requirements. Furthermore, we couple it with a method to utilize the per-class queuing system to reserve bandwidth for the transmitted video to optimize QoS/QoE and enhance the overall resource usage.
4. The thesis methodology was applied by streaming videos of different resolutions and evaluating their quality performance under a network topology experiencing packet loss and congestion.

1.6 Research Scope

In this study, we focus on dynamically placing video flows on guaranteed QoS routes and reserving resources that satisfy several individual QoE metrics, namely packet loss and bandwidth of HD and SD video stream resolutions.

1.7 Thesis Outlines

This thesis is organized into six chapters as follows:

Chapter One: Introduction of the thesis including background and research motivation, problem statement, research hypotheses and objectives, the research methodology contributions, research scope, and thesis outline.

Chapter Two: Describes the background technologies of the research area. Moreover, it presents an understanding of video QoS/ QoE. Different approaches and techniques for video QoE measurement are discussed. The video coding techniques, codecs, and network protocols employed in this thesis are also presented.

Chapter Three: related works, discuss current research works in the video streaming QoS routing and resource reservation.

Chapter Four: methodology, presents the architecture and overview of the proposed solutions.

Chapter Five: presents an experimental evaluation of the proposed model, and result analysis of the objective and subjective video quality.

Chapter Six: Presents the conclusions and future works of the thesis.

CHAPTER II

BACKGROUND

2.1 Introduction

This chapter presents the background technologies related to this study. It includes a general overview of the video streaming architecture and its protocols. In addition, this chapter provides a basic understanding of video quality of experience and quality of service. Streaming quality parameters, subjective and objective QoE measurement methods, and metrics are also discussed. Next, the theoretical background of the software-defined network is explained, and a brief description of the SDN building blocks is given. In addition, the OpenFlow protocols and OpenFlow QoS are demonstrated.

2.2 Video Streaming Service and Protocols

Streaming video delivery attempts to resolve problems associated with downloading, while also providing numerous additional features. The basic idea of video streaming is to split the video into parts, transmit these parts one by one, and allow the receiver to decode and play the video as soon as these parts are received without having to wait for the delivery of the whole video. Upon request from the end user, the video content is retrieved from the server and the channel encoder adjusts the video stream to meet the QoS requirements of the network. Then the encoded video stream is split into packets and transmitted over the network. On the end user's device, the received digital data is converted in the source decoder into a continuous waveform that can be viewed by various players at the application layer (Alreshoodi, 2016).

A major problem with Internet best-effort service is the lack of QoS provisioning on the network, which is essential for video streaming. This is because video streaming requires guaranteed performance, which imposes special requirements from the networks, as this type of applications is characterized as real-time (strict latency) and rate-critical (certain data rate) (McCabe, 2010). Because of these special characteristics of video streaming, networks must be managed to provide an adequate and guaranteed level of QoS and QoE. Figure 2.1 gives an example of the video streaming session.

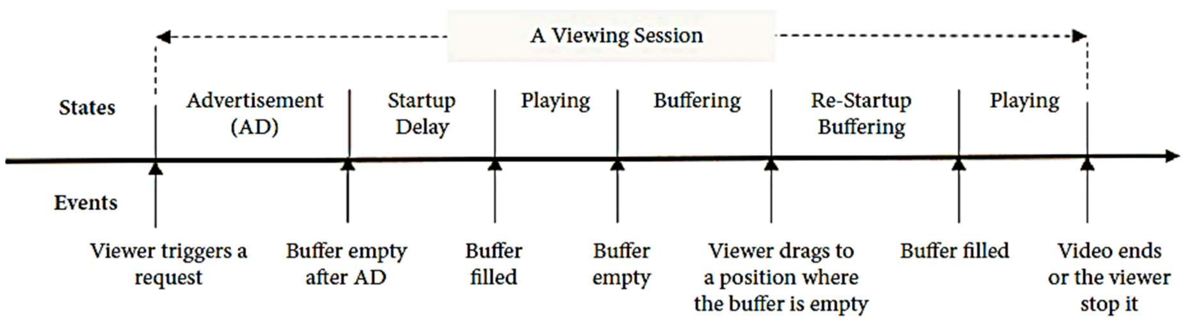

Figure 2.1: Video Streaming Session (Tan et al., 2018).

Transmission Control Protocol (TCP) and User Datagram Protocol (UDP) are the most popular lower-layer transport protocols. The TCP protocol is a connection-oriented protocol that offers a three-way handshake, retransmission, and error detection, but these features introduce a critical delay that is unacceptable for real-time applications such as video streaming. Although UDP is a connectionless and unreliable protocol, most real-time video services use UDP because it is faster than TCP, since it does not perform retransmission and error recovery. However, this causes more packet loss with UDP, which can corrupt the video content (Kaur et al., 2016). The next section focuses in particular on the Real-Time Transport Protocol (RTP) (Schulzrinne et al., 2003), and the Real Time Streaming Protocol (RTSP) (Schulzrinne, et al., 1998) are briefly introduced since they were mainly used in this work.

RTP and RTSP are commonly used for real-time application delivery. RTP is transmitted over UDP and works in conjunction with Real-Time Control Protocol (RTCP), which operates at the session layer. The main function of RTCP is to monitor data transmission. Detection of lost packets is possible with RTP.

RTSP is a network control protocol designed for use in entertainment and communication systems to control streaming media servers. It establishes and controls media streams between devices and servers by acting as a network remote control. RTSP supports multiple control requests (also referred to as "commands") such as play, pause, setup, etc. RTSP allows the selection of a transmission channel (e.g. UDP, multicast UDP, or TCP). RTSP uses RTP in conjunction with RTCP to transmit video and audio data.

2.3 Video Streaming QoE and QoS

The effective way to evaluate the quality of the video is by considering user perspective and assessment, this concept is known as quality of experience(QoE). ITU-T G.1080 defined QoE requirement as “The overall acceptability of an application or service, as perceived subjectively by the end-user” (ITU, 2008). The QoE is achieved by implementing subjective tests such as the mean opinion score (MOS). Although, QoE usually measured by objective QoE measurements metrics, which are algorithms designed to assist in the prediction of how actual viewers would estimate video quality. These objective metrics are influenced by QoS parameters and metrics.

In general, ITU-T E.800 defines the quality of service (QoS) as a set of characteristics that gives the telecommunications or network service the ability to satisfy the declared and implicit requirements of the user (ITU-T, 2008).

To satisfy the video streaming users’ and applications’ QoS needs; the network service must provide and meet a certain level of performance to guarantee their QoS. However, the expected levels of performance differ according to the user, application, and network limitations (Alreshoodi, 2016) (Wang and Crowcroft, 1996) (Mok et al., 2011). Therefore, the QoS model planned for video streaming must provide priority and reserve resources for this type of traffic so end users get acceptable QoE.

2.3.1 Video Streaming QoS Parameters

Many primary parameters have an impact on the QoE of video streaming, these must be observed in order to provide better overall QoE, they can be categorized based on application level or based on network level performance metrics see Figure 2.2.

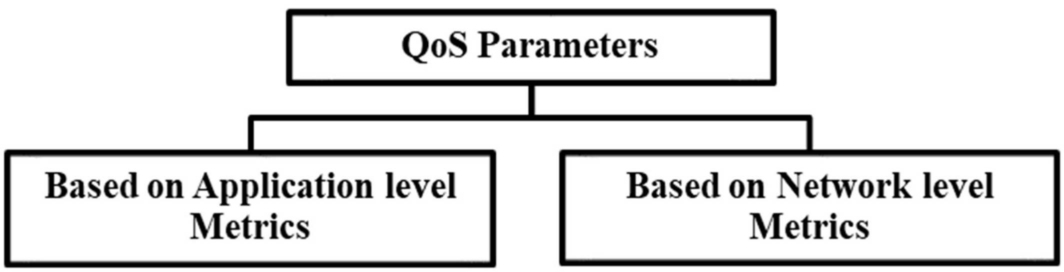

Figure 2.2: The Parameters Influence QoE of video streaming.

During the transmission of the video from the server to the client, its quality is distorted by the operations that are performed on it by the video encoders and video decoders, or by network conditions. Figure 2.3 shows the video transmission path where the original video send by media server to its clients, the received video called distorted video. Moreover, the figure illustrates the transmission points where QoS may be degraded, such as video encoder/decoder or at the network level.

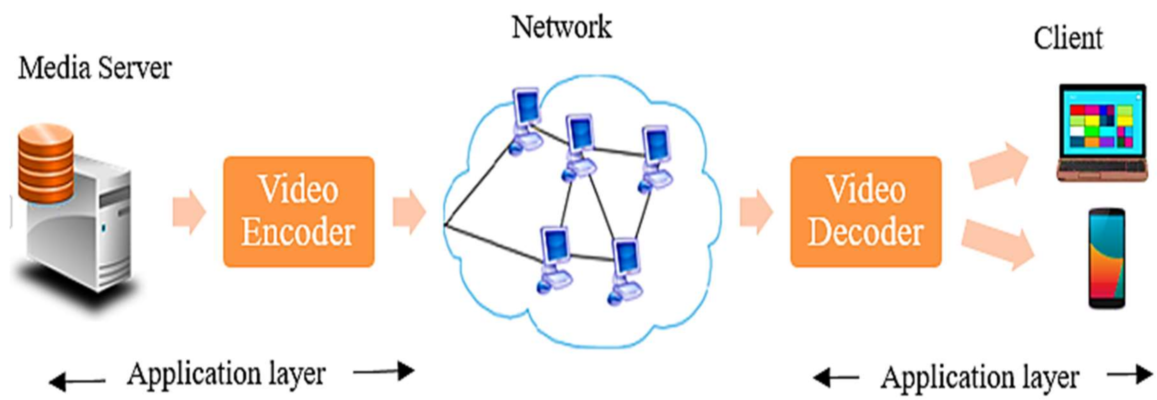

Figure 2.3: The Video Transmission Path.

To measure the QoE, the original video is compared to the distorted video received by the end viewers. Due to compression and block-based coding schemes and network state, the video may suffer from the following compression artifacts or visual distortions (Chen et al., 2014):

- Edginess is variations that occur at the image edges between the original and distorted video.
- Blockiness or Blocking effect refers to coding errors caused by block-based coding schemes such as H.263, H.264, and MPEG-4. It happens because image blocks are coded separately from each other, which results in visible boundaries between adjacent coding blocks.
- Blurriness or fuzziness results from loss of spatial information or edge sharpness, which makes objects appear out of focus.
- Motion jerkiness refers to perceived non-smooth video during playback, due to reducing frame rates or frame droppings that come from transmission errors or network jitter.

2.3.1.1 Video Streaming QoS Parameters Based on Application Level Metrics

Application level QoS parameters are media-related and content-related performance parameters, which are associated with video coding and compression techniques.

Video Encoding/Decoding is a mechanism performed by a software called codecs, the coder compresses the original video before transmission on the network to create a manageable and smaller stream. While the decoder decompresses the video after it is received by the target software. The encoded video is received in multimedia containers such as MP4, WebM, and Ogg. These containers consist of metadata, subtitles, video, and audio stream. The metadata includes information about the encoding parameters such as bitrate, image resolution, frame rate, and the codec that will be used to decode the received video. The MPEG-2, MPEG-4, and H.26x are recognized video CODECs and compression schemes (Chen et al., 2014) (Al Hasrouty, 2018). The codecs used to encode the video streams analyzed in this research are H.264/AVC and H.265/HEVC.

H.264/MPEG-4 AVC is a video coding compression standard developed by a team from the Video Coding Experts Group (VCEG) of the ITU-T and the Moving Picture Experts Group (MPEG) of ISO/IEC, also known as MPEG-4 Part 10 or H.264/Advanced Video Coding (AVC). It uses a MPEG-4 compression mechanism for a variety of applications and videos such as network video streaming, video conferencing, low/high resolutions, and DVD (Digital Video Disc) storage (Sullivan et al., 2004). H.264 is lossy compression technology which may lead to video quality degradation. Moreover, H.265/HEVC (High-Efficiency Video Coding) is a successor to H.264, developed by the same team. HEVC is designed to achieve a 50% bit-rate reduction with the same level of video quality compared to previous standards. Besides, it supports a different range of resolutions including ultra-high definition (8K UHD) (Sullivan et al., 2012).

Video quality of experience (QoE) is influenced by video compression formats and their complexity, some of the important video coding-related QoS factors are briefly explained below:

- Temporal and spatial features of the video are a compression technique to reduce redundancies in video frames. In temporal compression, the encoder encodes the key frames (called delta frames) only, besides sending pixels that change between successive frames or images, rather than encoding the complete images repeatedly. While spatial compression is redundancy within a single image or frame, in other words, it reduces the repeated pixel values inside a frame. Therefore, if the video is dynamic with changing colors and many motions then it may be vulnerable to jitter and packet loss because much information will get lost (Chen et al., 2014).
- Bitrate is the number of bits per second generated by the video encoder. Video codec encodes video at a Constant Bit Rate (CBR) or a Variable Bit Rate (VBR). CBR is most commonly used for streaming video and it keeps the bitrate identical during the encoding process. However, CBR could degrade the quality because it might not allocate enough bits for the complex sections of the video. On the other hand, VBR can solve the problem by assigning different bitrates, higher bitrates for complex segments, and less rate for less complex parts. Yet, it takes a longer time to encode videos due to the complexity of its process. In general, the higher the bitrate increases the video QoE (Alreshoodi, 2016) (Chen et al., 2014).
- Frame Rates (FPS) refer to the frequency of consecutive images called frames presented per second. Higher frame rates lead to better video QoE, however with a given fixed

encoding bitrate higher frame rate means a lower number of bits for each frame, which lead to coding and compression distortions, as a result, influence viewing experiences (Alreshoodi, 2016) (Chen et al., 2014).

- Video resolution is the number of points (pixels) in the frame, stated as width by height such as 640×360 and 1280×720, also refer to it as format or screen size. Usually, video resolution is shortened by its height, for example, 1080p rather than 1920p×1080p. There are three types of video resolution, Standard Definition (360 and 480 SD), High Definition (720 HD Ready or 1080 Full HD), and Ultra HD (4K). SD quality looks blurry and much less defined compared to HD which is much clearer. Ideally, a higher frame resolution produces a better video quality, however, this does not guarantee the QoE, since this depends on network bandwidth, network conditions, and the codec device's processing power (Alreshoodi, 2016) (Chen et al., 2014).
- The Quantization Parameter (QP) determines the amount of temporal and spatial information details that will be saved during image block compression. A large value of QP means high compression, as result decreasing of the output bit rate at the expense of quality loss (Gries et al., 1996).

2.3.1.2 Video Streaming QoS Parameters Based on Network Performance Metrics

Nowadays, video streaming is turn out to be an essential service on the Internet and it witnesses greedy usage by the end users. Besides, the development of video resolutions requires high network capacity as result networks must provide more bandwidth to guarantee the quality of service. However, the best effort network available bandwidth changes according to network failures. Bandwidth variations and unreliable transmission could lead to packet loss, delay, and jitter, all of these parameters impact the video streaming sessions QoE (Yu et al., 2015) (Alreshoodi, 2016) (Nam et al., 2014) (Chen et al., 2014).

ITU categorizes the video streaming applications under class 4 of QoS classes as listed in Table 2.1. The next points briefly present these network transfer performance parameters.

Table 2.1 IP Network Performance QoS Class for Applications (ITU-T, 2011).

Network	Class 0	Class 1	Class 2	Class 3	Class 4	Class 5
IP Packet Transfer Delay (IPTD)	100ms	400ms	100ms	400ms	1s	ND
Jitter (IP Packet Delay Variation (IPDV))	50ms	50ms	ND	ND	U/D	ND
IP packet Loss Ratio (IPLR)	10^{-3}	1x10-3	1x10-3	1x10-3	1x10-3	ND

- The bandwidth of a network indicates how much data can be transmitted in a given amount of time, usually measured in bits per second (bps). Streaming video is a resource-intensive process and consumes a lot of bandwidth on networks. As a result, streaming video over networks with limited bandwidth can be challenging. Even with the popularity of video streaming over the Internet, server and network bandwidth have become critical limiting factors because of the high bandwidth requirements and persistent nature of video streaming. In streaming quality, bandwidth consumption is the most influential factor because network congestion occurs when a link or node is overloaded, resulting in packet loss and increased delay. So reducing bandwidth consumption increases network performance in terms of delay, jitter, and packet loss (Van et al., 2002) (Al-Madani et al., 2014). Moreover, in terms of QoE, bandwidth is the most important parameter. It depends on several parameters, mainly the codec, the frame rate, and the image size (resolution).

According to Netflix, SD quality should be delivered at a rate of 3.0 Megabits per second, HD quality should be delivered at a rate of 5.0 Megabits per second, and Ultra HD quality should be delivered at a rate between 15 to 25 Megabits per second (Netflix, 2020).

- Packet loss probability or ratio (IPLR) is defined by ITU-T as the ratio of the total lost IP packet to the total number of packets transmitted, ITU defines 10^{-3} (0.001) as an acceptable loss ratio for video streaming (Seitz, 2003), however with the evolution of video data rate this ratio might be damaging. Furthermore, Cisco recommends that in order to get an acceptable QoE, the average loss thresholds should not be exceeded 0.5% for SD and 0.05% for HD (Cisco, 2013). High packet loss rates result in lower video QoE, where the frame may freeze during playback due to packet loss and then jump to the next arrived frame, also, losses of two consecutive video frames would be observed by most users (Shi et al., 2009) and could be destructive to the process of reconstructing the video image. Further, the higher video bitrate as in the HD means more packet loss. There are many sources of losing data packets, like network conjunction, poor connection reliability, protection switching mechanisms such as Multi-Protocol Label Switching (MPLS), security threats, the intentional packet loss that is used as a technique in network management to balance or prioritize available bandwidth and non-congestion related packet loss or random losses in Wi-Fi Networks. The packet loss can be decreased by retransmission, however, this increases the delay and jitter (ITU, 2008) (Alreshoodi, 2016) (Chen et al., 2014).
- IP packet transfer delay (IPTD) or latency is the time it takes for a packet to get from the sender to the receiver and it depends on the network capacity. ITU defines 1 second as an acceptable delay for video streaming (Seitz, 2003. Delay has a direct impact on user

preserved quality because a larger delay increases the frame start-up time in the video playback (Chen et al., 2014) (Kim and Choi, 2010).

- Jitter or IP packet delay variation (IPDV) is a measure of the difference between the absolute IP packet transfer delay and a defined reference IP packet transfer delay, between the sender and receiver, where the reference IP packet transfer delay between source and destination is the absolute IP packet transfer delay experienced by the first IP packet between them (Seitz, 2003). Video streaming requires low jitter otherwise most recent frame may freeze. jitter can be eliminated by buffering, although at the expense of increasing fixed delay (Chen et al., 2014) (Kim and Choi, 2010).

2.3.2 QoE Subjective and objective Measurement Methods and Metrics

Generally, video streaming QoE is measured by subjective or objective methods or both of them. Subjective metrics were conducted by asking subjects to rate the video they have been watching, the subjects rate the quality as excellent, good, fair, poor, or bad as their perception of QoE. The Mean Opinion Score (MOS) is a well-known QoE scale, which is an average of scores across subjects, the MOS maps the ratings between excellent and bad to numerical scores between 5 to 1 as shown in Table 2.2 (ITU-TP, 2008) (Streijl et al., 2016).

Table 2.2: Example of MOS Rating Scheme.

Score	**Quality**
Excellent	5
Good	4
Fair	3
Poor	2
Bad	1

However, subjective methods tests are time-consuming and costly (ITU, 2008). Therefore, objective methods could be used to predict the expected subjective QoE, by applying associated QoS metrics that have an impact on the video QoE, as a result, the objective test can correlate well with the subjective test results (ITU, 2008) (Alreshoodi, 2016). Moreover, depending on QoS layer architecture, objective tests are divided into network or application QoS performance metrics, Figure 2.4 illustrate a simple diagram of the objective methods.

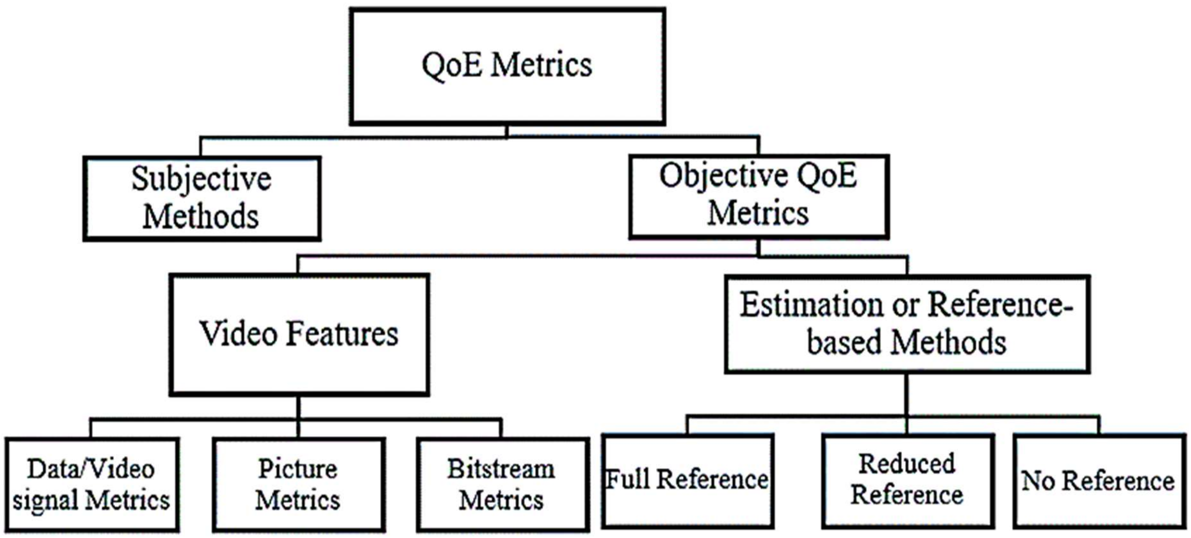

Figure 2.4: Classification of QoS and QoE Measurement Metrics.

2.3.2.1 Objective QoE Metrics

Objective measurements are mathematical models and computational algorithms have been developed to estimate the QoE based on QoS metrics (ITU, 2008). These metrics can be classified as estimation methods or based on elements constructing the video.

Firstly, according to the amount of reference information or source signal required to test the video quality, there are three categories of estimation methods (Jonnalagadda and Musti, 2012) (Hands et al., 2005):

- Full reference (FR): are methods access both the original -called reference-and processed video -called distorted- to assess the quality. The properties of the two tested videos are compared frame-by-frame to check different features such as contrast features, and color processing. FR is primarily for laboratory-based performance testing.
- Reduced reference (RR): in this model, the comparison is made based on some features extracted from the original reference signal.
- No reference (NR): this method does not depend on the original reference video, hence it is suitable for use in real-time quality monitoring and control.

Secondly, different approaches and models were developed to provide metrics for quality depending on the video features such as data or signal based also called pixel-based models, bit-stream based, and picture or image-based metrics (Winkler and Mohandas, 2008) (Chen et al., 2014):

1. In signal based or pixel-based, the video quality is measured by testing video images pixel by pixel. The two most known are mean squared error (MSE) and peak signal-to-noise ratio (PSNR). Although PSNR is popularly recognized due to its simplicity, however, these metrics are poorly correlated to subjective results (Chen et al., 2014).
2. Bitstream or parametric/bitstream-based metrics are concerned with encoded video bytes for the frame. They are intended to measure the impact of IP network layer level and video bitstream and lastly have been standardized by ITU (ITU-TP, 2017).
3. Finally, there are two approaches used in picture-based metrics, the vision modeling approach also called the psychophysical approach and the engineering approach. they are looking for the effects of specific content and distortions on video quality:

a. Vision modeling depends on features of the Human Visual System (HVS), such as spatial and temporal features, pattern masking, and contrast sensitivity. Moving Picture Quality Metric (MPQM) (Van den Branden Lambrecht and Verscheure, 1996) and Perceptual Video Quality Measure (PVQM) (Hekstra et al., 2002) are proposed under this classification.
b. While the engineering approach is based on analyzing and extracting of particular artifacts or features on examined videos, these image features can be structural image elements or distortion patterns like (edginess, and block artifacts). The Structural Similarity Index Metric (SSIM) (Wang et al., 2004) is one of the widely known engineering approaches, in addition to the Video Multimethod Assessment Fusion (VMAF) which was proposed by Netflix (Li et al., 2016).

The metrics based on video features also can be classified under reference-based classification methods, for example, PSNR and SSIM are FR metric techniques, because they require original and distorted videos to accomplish the quality assessment. On the other hand, bitstream methods are NR models and do not need access to the source video.

In summary, various types of objective models are known, including statistical models, network planning models, and arithmetic models. This thesis mainly uses two types of full reference models measurement metrics for QoE estimation:

- SSIM is a statistical model, which measures the similarity between the original video and the distorted video for each frame and merges the output into a distortion map. SSIM is designed to be correlated with the quality perception of the human visual system hence it higher predictive quality value to the subjective tests than PSNR.

- Lastly, VMAF is a powerful method focused on quality degradation due to compression artifacts and scaling artifacts, it uses machine learning techniques to evaluate the perceptual video quality.

Table 2.3 shows an approximation of objective QoE scales for PSNR, SSIM, and VMAF video quality metrics mapped to the subjective QoE MOS scores (Khan et al., 2008) (Zinner et al., 2010) (Li et al., 2018).

Table 2.3: PSNR SSIM and VMAF values mapped according to MOS scores.

MOS	PSNR	SSIM	VMAF
Excellent = 5	> 37	> 0.99	80 -100
Good = 4	31 – 36.9	≥ 0.95 & < 0.99	60 - 79
Fair = 3	25 – 30.9	≥ 0.88 & < 0.95	40 - 59
Poor = 2	20 – 24.9	≥ 0.5 & < 0.88	20 - 39
Bad = 1	< 19.9	< 0.5	< 20

2.4 Software-defined Network History and Building Blocks

Recently, the IP Networks covered wide areas of the world, however, this expansion added a lot of complexities at the expense of management, control, and the routed data in the IP networks, because of the extreme distribution of the networking devices and management servers' applications. Moreover, existing IP networks combine the control decisions (control plane) and the forwarding (data plane) traffic in network devices: such as routers, switches, and other devices. The control plane decides how to deal with the network traffic which represents the construction of routing tables, traffic engineering, QoS rules, enforcing security, policies, and so on. Whilst, the data plane is in charge of efficient forwarding of data according to commands of the control plane. The mixture of these planes and decentralized controlling networking devices inherits static and complex architecture, as well leads to difficulty in fault tolerance since to several devices' misconfigurations. Furthermore, it restricts innovation and development flexibility in the network software and hardware, due to the lack of vendor solutions for network management, and proprietary software (Kreutz et al., 2015) (Ranjan et al., 2014).

The software-defined network (SDN) is a new network technology proposed recently to solve the problems of controlling and management of the networks. It intended to separate the network's control and data (forwarding) planes to get better management and optimization, as a result, it makes the underlying network infrastructure abstracted from the applications and controlled via software-based controllers (Nunes et al., 2014). In Addition, the network turns out to be easier programmable through software applications and this is considered major a factor that distinguishes the SDN. However, according to the authors (Feamster et al., 2014), the idea of a programmable network emerged initially from the concept of active networking which began in the early to mid-1990s. The history of SDN began with active networks, then the appearance of the control and data plane decoupling concept in the period from 2001 to 2007, which bring into line the development of open interfaces between the control and data planes, and lastly, from 2007 to 2010 the development of network operating systems and OpenFlow API (Feamster et al., 2014). After that, the SDN began to attract the attention of the research community and Internet service providers.

2.4.1 Building Blocks

SDN separates the control plane from the data plane. The Control plane decides how to handle network the traffic while the data plane forwards traffic according to decisions the made by control plane. The SDN is the control data plane logically logical centralized controller (physically distributed), this controller performs all complicated functions of the network devices. SDN also focuses on the features of the open interfaces, where the controllers communicate with the data plane through southbound application programming interfaces (APIs), while alternate information with the application plane through northbound APIs. Figure 2.5 shows the SDN planes, while Figure 2.6 illustrates the SDN functional architecture abstractions and main building blocks.

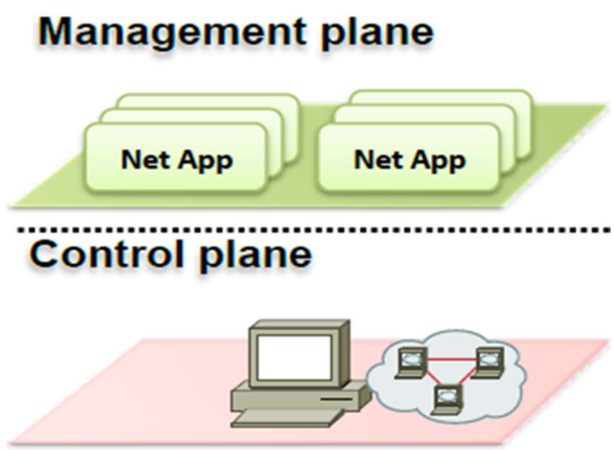

Figure 2.5: SDN Data, Control and Management plane (Kreutz et al., 2015).

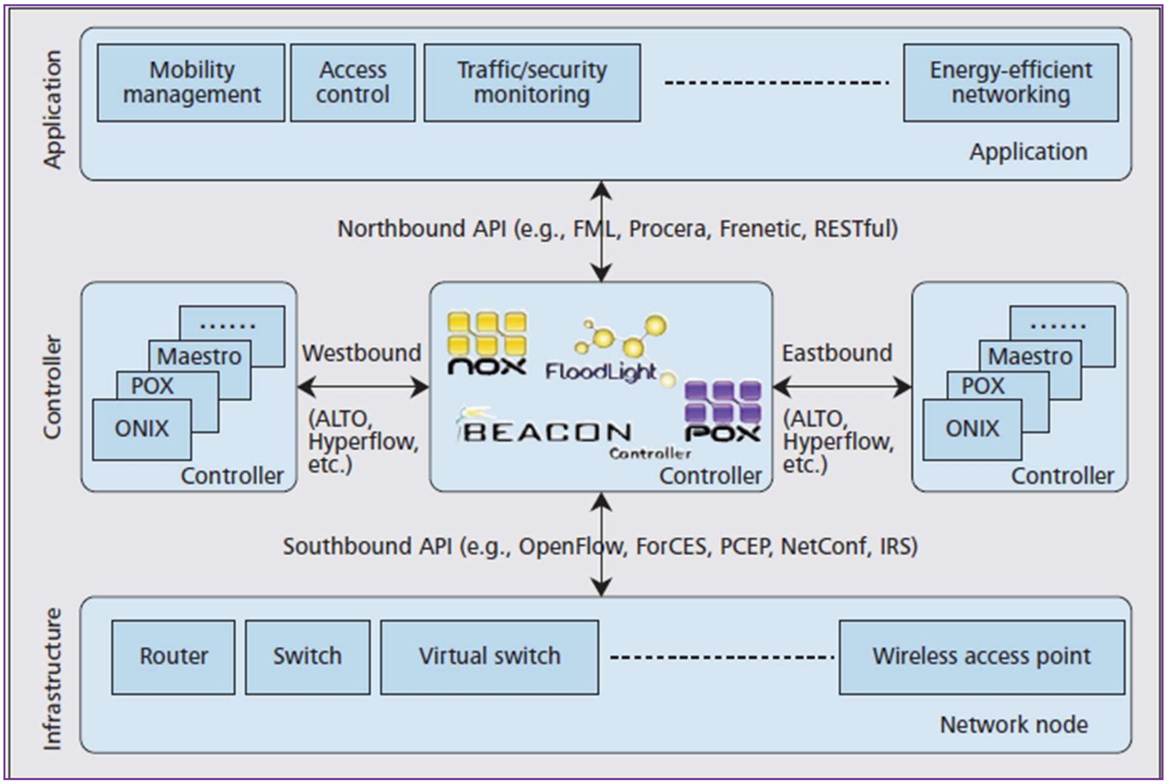

Figure 2.6: SDN Functional Architecture Abstraction (Sezer et al., 2013).

The main building blocks of SDN and OpenFlow Networks address as follows:

- **Control Plane** presents an abstract view of the complete network infrastructure. Many controllers Software are used in SDN such as NOX (first SDN controller), Floodlight, SANE controller, IRIS, and others. The controller is responsible to permit each network flow, if it permits a flow it computes a route or path of that flow, and adds an entry of it in each of the switches along the path (populated switches tables entries). In more clarity, if a flow arrives at a switch one of the next scenarios will be followed. Firstly, the switch lockup for it in the flow table, if there is no matching flow it encapsulates and forwards the first packet of a flow to the controller, which decides whether to add the flow in the switch table or not this is called reactive mode. Alternatively, in proactive mode, the switch does not forward the flow packet to the controller because all possible flow matches populate previously by the controller. After that, the switch forwards the incoming packets via the proper port based on the table entries, further, the SDN controller can order the switch to drop packets for security reasons or network performance management (ONF, 2014) (Stallings, 2013).

- **East-West Protocols** manage the communication and interaction between multiple controllers in the control plane.
- **Northbound Application Interfaces** offered by the network operating system to application developers, these APIs is an open source-based and their interface abstracts the low-level instruction sets of underlying network components (Kreutz et al., 2015) (Jammal et al., 2014).

- **Management Plane** contains the set of applications programmed by developers to manage the networks for example firewalls, access control lists, traffic engineering, load balancer, routing algorithms, monitoring systems, and so on. Those applications utilize the functions offered by the northbound APIs to implement network control policies and operation logic.
- **Data Plane** represents the forwarding hardware in the SDN network architecture such as routers, switches, and other devices (such as firewalls intruders' detection systems, etc.).
  - **Southbound APIs Protocols** are used by the controller to communicate with forwarding devices. These devices have well-defined instructions used to take actions on the incoming packets (such as forward to specific ports, drop, forward to the controller, and rewrite some header), these instructions are defined by southbound interfaces such as OpenFlow and ForCES protocols (Kreutz et al., 2015) (Jammal et al., 2014).
- **OpenFlow protocol** is based on an Ethernet switch, with an internal flow table, and a standardized interface to add and remove flow entries. It is an open protocol to program the

flow table in different switches and routers used to generalize SDN architecture. It allows controllers to direct access and manipulation the forwarding plane of network devices. An OpenFlow switch has one or more tables of packet-handling rules; each rule matches a subset of the traffic and performs certain actions (dropping, forwarding, modifying, etc.) on the traffic. These rules are installed by a controller, inside the OpenFlow switch and it can make the switch can act like a router, switch, firewall, or carry out other functions (McKeown et al., 2008). OpenFlow allows the controller to add, update, and delete flow entries proactively and reactively in flow tables. In Figure 2.7 it is shown that the OpenFlow protocols used in switching devices and controllers interface.

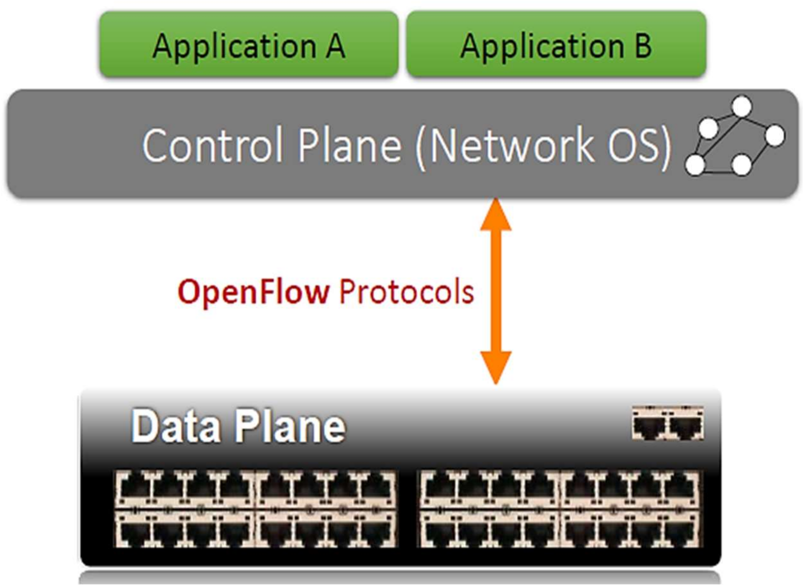

Figure 2.7: OpenFlow Protocols Act as Abstraction between Data and Control plane.

2.4.2 OpenFlow QoS

There have been some additions to OpenFlow over the years that allow the implementation of QoS frameworks. OpenFlow 1.0 introduced an optional action called -enqueue- that allows flow entries to forward packets over a queue on a port. Despite being able to query queue information, OpenFlow cannot configure them. As of OpenFlow 1.2, all queues in a switch can be queried simultaneously, and additional queue properties have been defined. In OpenFlow 1.3, rate-based packet remarking and rate-based rate limiting was added. Meter tables are used for this purpose which are made up of meter entries. It is possible to define multiple meter bands per meter entry. The meter bands consist of (a band type, a drop or DSCP remark, a rate (rate and burst), counters, and optionally type-specific arguments). The meter instruction has been replaced by a meter action in OpenFlow 1.5. The meter action allows for multiple meters to be attached to flow entries and for meters to be used in group buckets. A new feature in this version is the addition of egress tables that can be quite useful for the quality of service (Arreaza Govea, 2019).

CHAPTER III

RELATED WORKS

3.1 Introduction

This chapter presents SDN research related to the proposed solutions. Section 3.2 reviews existing QoS-based routing and bandwidth reservation solutions for video streaming. The section 3.3 analyzes and discusses related work approaches to the proposed solution (Video Streaming Adaptive QoS-based Routing and Resource Reservation (VQoSRR)).

3.2 A Review of Related SDN QoS Routing and Resource Reservation Mechanisms

Well-known techniques in the traditional Internet have been proposed to offer QoS in intra-domain and inter-domain networks, such as integrated services (IntServ) (Braden et al., 1994), differentiated services (DiffServ) (Blake et al., 1998), and Multiprotocol Label Switching MPLS (Rosen et al., 1999). However, their implementations are limited due to the decentralized control of the current best-effort networks, resulting in static and complex architecture where they cannot adapt to changing conditions on the network topology. Moreover, networking devices are more exposed to failures because of misconfigurations and missing automated settings (Civanlar et al., 2010).

On the other hand, several SDN routing QoS mechanisms have been proposed to support video streaming. H. E. Egilmez et al. designed the OpenQoS multimedia controller to provide a dynamic end-to-end QoS routing according to the network state (Egilmez et al., 2012). The controller enhances the video quality experienced by end-users by offering two paths. One is a QoS route specified for multimedia streaming, and the second one is the shortest path for the other data. The route calculation metrics are delay, congestion, and available bandwidth. However, the used delay is fixed, which means that the dynamic route is built only on the gathered bandwidth statistics.

The architecture proposed in (Civanlar et al., 2010) provides an analytical optimization of the QoS routing problem, which is based on linear programming for scalable video coding (SVC). It aims to minimize the delay and offers no packet loss for QoS traffic. Likewise, (Egilmez et al., 2011), and (Egilmez et al., 2013) proposed two optimization frameworks for SVC video stream and computed paths using the LARAC algorithm. The first study considers the SVC base layer as lossless flow with no packet loss while the enhancement layers could tolerate packet losses. In case of congestion, only the video base layer allows to reroute to other available non-shortest path routes. The second framework reroutes both base and enhancement layers to QoS-based routes, while the best effort traffic remains on its shortest path. These papers depended on the packet loss rate as a metric to determine the new routes. They assumed that if there is not sufficient available capacity, then, there will be packet loss. However, they do not consider the other possibilities of packet loss, such as security settings or attacks. They did not calculate the actual packet loss rate given by the obtained network state information. The study published in (Egilmez et al., 2013) used the delay variation as the constraint with packet loss rate to calculate QoS routes. Also, they ignored the influence of sufficient bandwidth as a routing metric. For video streaming, the delay variation could be improved by increasing the receiver buffers. Whereas, network bandwidth is often more critical for these applications, because if it is limited, this will lead to the queuing delay, and the packet loss rate will be higher, which will decrease the overall QoE of the end user.

However, Egilmez et al. (2011, 2012, 2013) proposals queried the network status each second. This would result in additional overhead for the controller to compute routing per network state, especially if the state had not changed significantly.

Yan et al. (Yan et al., 2015) presented the HiQoS SDN framework to guarantee QoS. It is based on differentiated services and computes multiple paths between the source and destination. The framework distinguishes between different types of traffic by source IP address and categorizes user traffic into a video stream, interactive audio/video, and best-effort flow. Additionally, a modified Dijkstra algorithm was used to select the optimal path for the new flow according to the lowest bandwidth consumption. Using multiple paths enables HiQoS to be resistant to link failures through rerouting flow to another path. However, HiQoS only used the minimal bandwidth utilization of a queue as a metric, ignoring other video streaming quality metrics. Also, it classifies the flow with the source IP address, this might be not precise for video streaming, since different videos with different quality demands may be originated from the same device. Our proposed VQoSRR, instead, employs queue mechanisms to meet bandwidth guarantees for video traffic in addition to providing two routing paths between the source and destination to satisfy multiple QoE constraints.

ARVS (Yu et al., 2015) suggested a two-level QoS routing approach for video streaming over SDN. They used the delay variation and cost to select the optimal paths for videos. Their work is like (Egilmez et al., 2013), except they utilized the shortest plus feasible QoS paths for all the traffic types. Both, the base layer video stream (level-1 QoS) and best-effort traffic are routed through the shortest path. ARVS will check this path against jitter constrain periodically, if it has jitter values over a threshold, the available bandwidth along the QoS path is examined, in case the bandwidth is sufficient, the level-1 flow has priority to be re-routed to this path, and enhancement layer packets (level-2 QoS) remained on the shortest path, while the level-2 re-routes to the QoS path and the base layer packets stay in the shortest path. However, packet loss is the most important constrained metric for a video stream at the network level and buffers at the application level can solve the jitter problem.

In the same direction (Xu et al., 2015), offered a QoS-enabled management framework to support a queuing mechanism and route optimization algorithm to guarantee the transmission of the flow over SDN networks. It provides a suitable QoS for video streaming and multimedia applications and also classifies the flows into different QoS levels. Their algorithm solves the constrained shortest path (CSP) problem based on the delay parameter. Besides, they compute the routing path cost with two additive metrics (delay and packet loss). The routing algorithm dynamically rerouted the high-priority flow when network congestion occurs; if there is no feasible path to transmit the QoS flow, the framework enabled a queue reservation instead. Owens and Durresi (Owens and Durresi, 2015). designed a video over software-defined networking (VSDN) architecture and protocol for optimizing QoS routing and queuing for video transmutation. They implemented a signaling QoS framework like integrated services (IntServ) to guarantee end-to-end QoS for video applications. With the VSDN protocol, video applications can request video service from the network by providing a QoS API used by the sender and receiver. Despite that, the architecture may not be scalable in large networks due to the potential signaling overhead between video senders and receivers.

To enhance network quality of service, Sendra et al. (Sendra et al.,2017) presented a routing optimization system for SDN by applying reinforcement learning (RL) to enhance network

QoS. The proposed routing protocol used this artificial intelligence (AI) method to select the optimal paths with the least cost in terms of delay, loss rate, and bandwidth according to the network status. Additionally, Guo et al. (Guo et al., 2021) applied reinforcement learning (RL) in hybrid SDNs. It achieved link load balancing with the avoidance of routing loops by responding to dynamically changing traffic. Moreover, in (Liu et al., 2021) the authors used a deep reinforcement learning (DRL) scheme to optimize routing in the SDN of data-center networks. The method combines different network resources, such as bandwidth and cache memory, in order to minimize the delay. Then, it uses this information to improve the routing performance.

Rego et al. (2017) analyzed the effect of the OSPF protocol on network quality. Their study compared the protocol performance of SDN and traditional networks. Another work, (Rego et al., 2019), took advantage of the SDN and AI to propose a dynamic routing metrics calculation for multimedia data. The presented proposal modified the OSPF protocol equation by considering serval metrics (bandwidth, delay, and loss). Besides, they implemented a messages exchange protocol between controller and switches to adapt the metrics according to the current topology state. Their results show that the delay and packet loss decreased, while bandwidth utilization increased. Also, a performance enhancement framework for IP video surveillance (IPVS) is presented in (Go et al., 2019). It takes advantage of SDN to adjust QoE bitrates and reroute traffic to maximize the resources available on the network. Employing rerouting ensures choosing the shortest and less loaded paths, so the video destination received the high-priority streams at superior quality. Framework results minimize packet loss, jitter, and latency, while it is also optimized the throughput.

Yamansavascilar et al. (Yamansavascilar et al., 2020) proposed a dynamic fault tolerance solution to improve the QoE of video streaming. They detect the congestion in the SDN link layer by using bidirectional forwarding detection (BFD) protocol, then, they employ a data plane link failure tolerance mechanism to find alternative paths in order to offer good QoE.

Furthermore, many other solutions have also been proposed for enabling video QoS in SDN networks. Ghalwash and Huang (Ghalwash and Huang, 2018) proposed a framework for applying QoS in an SDN-based network. For QoS, they select the traffic route based on the shortest end-to-end delay metric, using the Dijkstra algorithm. The framework monitored the port utilization to reduce congestion and packet loss. Three types of applications, namely TCP, UDP, and VoIP, were evaluated based on reduced delay, jitter, and packet loss.

Yu and Ke (Yu and Ke, 2018) presented a genetic algorithm-based routing method to provide efficient video delivery over SDN, called GA-SDN. It used chromosome fitness as a metric to choose the best path from multiple possible solutions. In case of congestion, the link weight will be increased to decrease fitness. Since their approach defined fitness as the reverse of path cost, when there was a higher fitness, it indicated a higher QoS. Similarly, Parsaei et al. (Parsaei et al., 2020) proposed a model for critical delay-sensitive telesurgery applications based on SDN networks. The model used a type-2 fuzzy system (T2FS) and cuckoo optimization algorithm (COA) to generate reliable QoS routes. The model computed two paths, with the first set as the primary path and the other an alternative in case of failures; the delay metric was used as a link constraint to determine which routes are optimal between the remote surgeon and the operating robots at the patient's side.

Henni et al. (Henni et al., 2020) developed a framework for QoS routing in SDN to enhance video streaming quality and best effort flow throughput. The authors leveraged SDN properties to create a consistent view of the network, consistent decisions, and a consistent enforcement strategy of rules. Their approach minimized the concentration of video streams on links to reduce packet loss and maximize QoS. Volpato et al. (Volpato et al., 2017) introduced an architecture that integrated autonomic and proactive QoS management into SDN environments. This architecture enabled QoS configuration on data plane devices. In addition, it monitored, predicted, and analyzed the network performance to achieve resources' optimizations and avoid degradations in the QoS. However, it focused on optimizing resource utilization without making any guarantees of meeting the service thresholds.

Sharma et al. (Sharma et al., 2014) proposed a framework to enable QoS for business customer traffic and provide on-demand prioritization based on flow differentiation and resource reservations. Flows are classified based on the type of service (TOS) field and the destination IP to differentiate best-effort traffic from business customer traffic. The authors applied the rate shaping technique to reserve queues, and they configured each router with a high priority queue and a low priority queue. In addition, the FlowQoS (Seddiki et al., 2014) system utilized SDN to provide per-flow QoS for broadband access networks. FlowQoS depended on classification and rate shaping based on the policies defined by the user. This system created a virtual switch topology inside the router and configured each switch according to a user-defined rate. Similarly, Khater and Hashemi (Khater and Hashemi, 2018) implemented differentiated services on SDN networks to enhance the quality of service. They distinguished the flow within the network by changing the differentiated service code point (DSCP) value in the ToS field and then assigning different queues to each flow based on the DSCP value. When necessary, their proposed method shifted the flows between switch queues to prevent increasing delays, while utilizing available capacity within other switch queues to accommodate new flows.

Rego et al. (Rego et al., 2018) described an architecture for monitoring urban traffic in emergencies based on SDN. Their approach combined SDNs and Internet of Things (IoT) networks for more effective management of emergency resources. Their architecture enabled the modification of vehicle routes dynamically by changing traffic lights to facilitate the movement of emergency service units. Canovas et al. (Canovas et al., 2020) proposed a multimedia traffic management system based on the QoE estimation scheme and traffic pattern classification for SDN networks. They implemented two models. The first one is a QoE model based on Bayesian regularized neural networks (BRNN) for multimedia traffic classification based on the objective QoE. A second model determines which video characteristics should be changed to improve QoE in difficult situations. These characteristics are selected based on QoS parameters.

3.3 Comparison and Discussion of Related Studies

Most of the previously mentioned studies focused on the CSP problem that tries to find the least cost path which satisfies only one constraint while ignoring the multiple constrained shortest path (MCSP) problem. MCSP problem aims to find a path subject to multiple constraints. Video flow is one class of service that requires combinations of routing metrics to realize its QoS. Therefore, CSP does not estimate or guarantee that the path could fulfill all QoS constraints individually.

Additionally, according to the previously mentioned related works the majority of them are focused on achieving only one constraint of video streaming (Civanlar et al., 2010), (Egilmez et al., 2013), (Yu et al., 2015) and (Xu et al., 2015), while some others are focused on optimizing the performance without addressing specific QoS requirements (Yan et al., 2015) and (Rego et al., 2019). Similarly, many of them use the cost of route selection based on a single mixed metric, and, as we stated previously when it depends on the SMM, it does not guarantee each QoS parameter individually. Alternatively, our proposal meets multiple individual QoS metrics.

Moreover, the effective way to evaluate the quality of the video is by considering the user perspective and assessment, which is known as QoE. It is achieved by implementing subjective tests or by objective QoE metrics. However, three studies (Yan et al., 2015; Xu et al., 2015; Rego et al., 2019) evaluated the video quality based on network-level performance measurements. Additionally, (Civanlar et al., 2010; Egilmez et al., 2013), and (Yu et al., 2015) measured the video quality via Signal-to-Noise Ratio (PSNR), but it does not reflect human perception. For this reason, it cannot be considered a reliable method for assessing QoE (Huynh-Thu and Ghanbari, 2012). Instead, we use SSIM and VMAF, which can analyze blurring, luminance, contrast, global noise, as well as blocking and color distortions to detect artifacts that can be perceived by human eyes. Table 3.1 illustrates a comparison between the proposed approach and related proposals in terms of a number of QoS-Based routing algorithms, video constraints, and QoS/QoE measurement metrics.

Table 3.1: Comparison of Related Work with the Proposed Solution on QoS-Based Routing Algorithms.

#	QoS-Based Routing Algorithm	Routing Metrics	The applied Constraints	Video Parameters	QoS/QoE Measurement metrics
Civanlar (Civanlar et al., 2010)	The Solution not Developed	Available Bandwidth	Delay	SVC B/E layer	PSNR
Egilmez (Egilmez et al., 2013)	LARAC	Two mixed metrics: Packet loss and jitter	Jitter	SVC B/E layer	PSNR
Yan (Yan et al., 2015)	Modified Dijkstra	Two mixed metrics: Delay and Bandwidth	Minimal Bandwidth Utilization	N/D	Server Response Time and Throughput

Yu (Yu et al., 2015)	LARAC	Two mixed metrics: Jitter and Packet loss	Jitter	N/D	PSNR
Xu (Xu et al., 2015)	Random Discretization Algorithm	Two mixed metrics: Packet loss and Delay	Delay	N/D	Packet Loss, Jitter, and Throughput
Rego (Rego et al., 2019)	Dijkstra	Three mixed metrics: Packet loss, Bandwidth, and Delay	AI Dynamical Cost Equation to Specify the Constraint between Loss, Bandwidth, and Delay	N/D	Bandwidth Utilization, Packet Loss, Jitter, and Delay
Our proposal	TwoLLWPs (See chapter IV)	Multiple Individual metrics: Available Bandwidth and Packet loss rate	Multiple individual Metrics: Bandwidth and Packet loss rate	SD, HD Ready, HD	SSIM, VMAF, and Throughput

On the other hand, many related works are focused on per-flow QoS routing for example (Ghalwash & Huang, 2018); (Egilmez et al., 2011, 2012, 2013); (Sendra et al., 2017) and (Yu & Ke, 2018). In contrast, others focused on resource reservation schemes only (Volpato et al., 2017); (Sharma et al., 2014); (Seddiki et al., 2014); and (Khater & Hashemi, 2018). Other approaches used both methods interchangeably (Xu et al., 2015) or together (Owens & Durresi, 2015) and (Yan et al., 2015). Using QoS-based routing only determines the path with the best chance of acquiring the requested QoS. However, it does not involve a mechanism to reserve the required resources (Crawley et al., 1998). It also imposes an overhead because it gathers traffic status information actively. Oppositely, using resource reservation alone provides a mechanism for reserving network resources. Although, it does not provide a method for determining which network path has sufficient resources for the requested QoS (Crawley et al., 1998). In addition, most studies focus on enhancing network performance without considering video streaming thresholds to ensure QoS/QoE requirements are maintained.

In our approach, instead, video flows are dynamically placed on QoS guaranteed routes that meet multiple individual QoE metrics, while reducing the overhead of obtaining network status by pre-computing alternative paths. Additionally, we reserve resources for network flows according to their importance to avoid degradation of video stream quality, especially during high network loads. A comparison of the proposed approach with other SDN QoS architectures in terms of QoS and reservation guarantee is shown in Table 3.2.

Table 3.2: A Comparison of some SDN QoS Architectures Related Research with the Proposed VQoSRR.

Techniques	**QoS Solution**	**Video Threshold metric**	**Video QoS/QoE Parameters**	**Flow Resource Management Model**
Ghalwash and Huang (Ghalwash & Huang, 2018)	SP	Not used	Not used	Not used
Egilmez et al. (Egilmez et al., 2011, 2012, 2013).	QR	Jitter: Guaranteed	Bitrate	Not used
Volpato et al. (Volpato et al., 2017)	DRE	Bandwidth, Loss, Latency: Optimized	Not used	Differentiation by Transport Port Address and protocol. Queues are provided according to the Knowledge Base context.
Sharma et al. (Sharma et al., 2014), Seddiki et al. (Seddiki et al., 2014)	DRE	Not used	Not used	Queue reservation with differentiating service based on IP header.
Khater and Hashemi (Khater & Hashemi, 2018)	DRE	Not used	Not used	Queue reservation with differentiating service based on DSCP.

Xu et al. (Xu et al., 2015)	QR or DRE	Delay: Guaranteed	N/D	Queue reservations with differentiating services based on a different level of priority.
Owens and Durresi (Owens & Durresi, 2015)	QR+RE	N/D	Video Resolution	Queue Reservation similar to IntServ.
Yan et al. (Yan et al., 2015)	SR+DRE	No metric used	N/D	Queue Reservation with differentiating service based on source IP Address
VQoSRR (proposed)	QR+DRE	Bandwidth and Packet Loss Rate: Guaranteed	Video Resolution	Queue reservation with differentiating service based on DSCP.
Notes	• SR: Shortest Path Routing • QR: QoS-based Routing based Solution. • RE: Reservation-based Solution.			• DRE: QoS Differentiation and Reservation-based Solution. • N/D: Not Defined.

CHAPTER IV

METHODOLOGY

4.1 Introduction

This chapter introduces and discusses the proposed Video Streaming Adaptive QoS-based Routing and Resource Reservation (VQoSRR) framework, it investigates the QoS-based routing and resource reservation and shows how their combination could achieve overall enhancement in QoS and QoE of video streaming. In order to demonstrate this, the general methodology phases are illustrated in Figure 4.1. The first section describes the Network model briefly, then the rest of this chapter presents the strategies and methods used to propose VQoSRR routing and reservation, followed by evaluation and validation metrics and techniques.

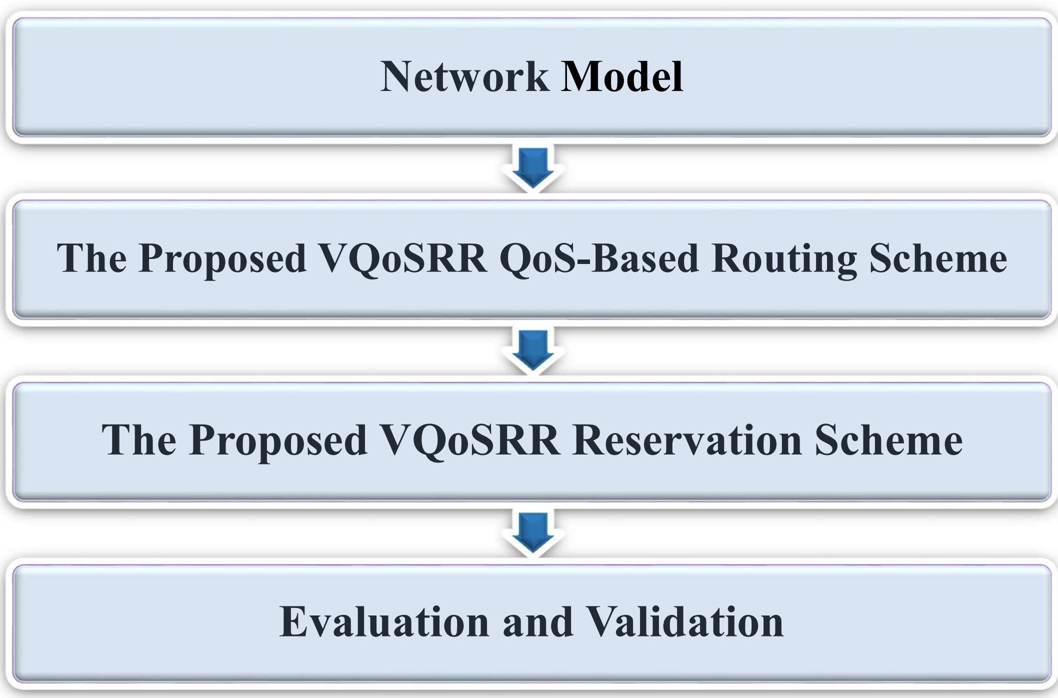

Figure 4.1: Research Methodology Phases

4.2 Network Model

The general proposed SDN network model and design components are illustrated in Figure 4.2, where the network is represented as three layers, application plane, control plane, and data plane. This section will describe these components briefly.

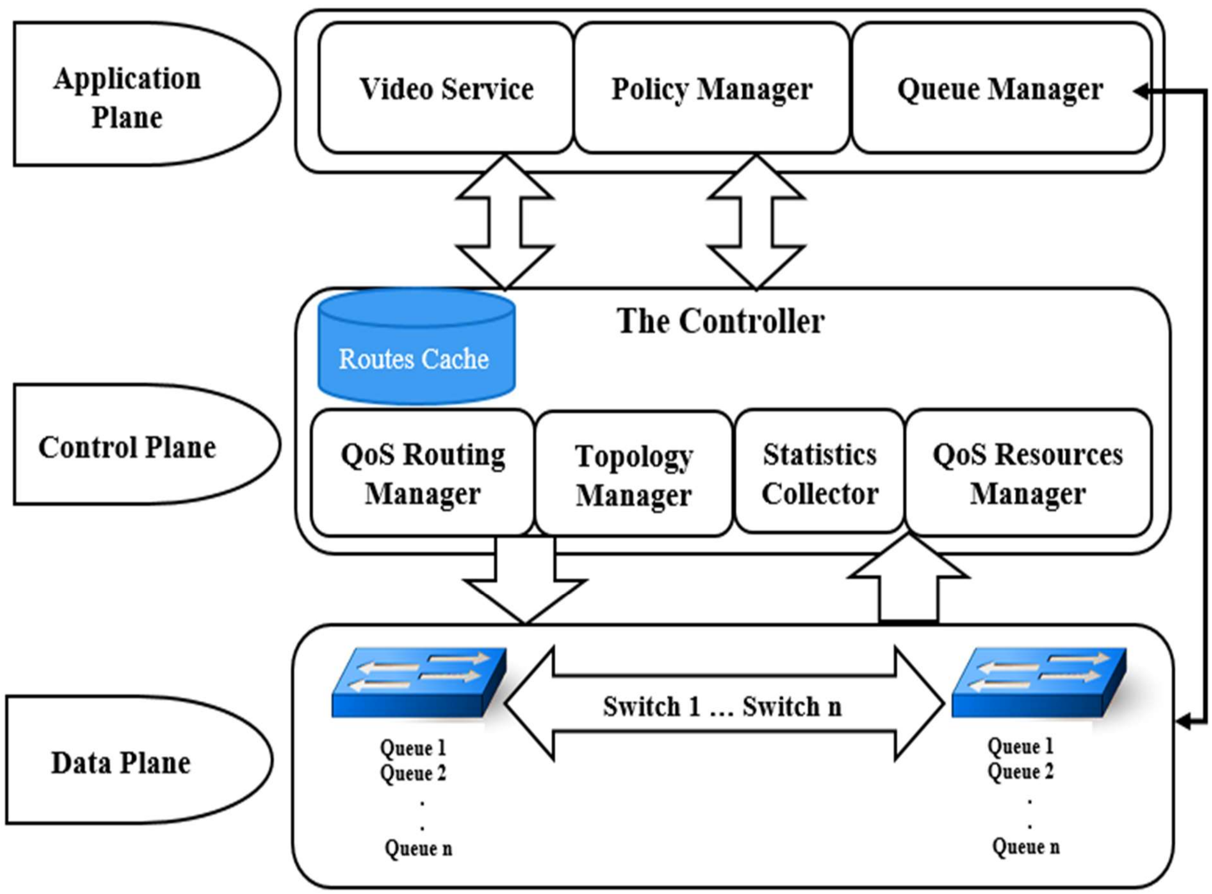

Figure 4.2: The VQoSRR SDN Architecture

4.2.1 Application Plane

1. **Video Service:** today's video streaming services and related industries continue to grow at an incredible rate, such as YouTube and Netflix. In addition to increasing their consumers and diversity of used devices to watch videos on the internet. This service requires acceptable

quality of experience to gain the satisfaction of its users. Furthermore, the current best-effort networks are not suitable to provide good QoS and QoE. Therefore, this thesis seeks to improve the QoE by enhancing the network QoS. This work evaluated the video performance using two different video formats: standard definition and high definition (HD ready falls into this category), all encoded with H.264/AVC.

Moreover, the video service applications (both client and server) were developed in Python. The TCP protocol was used to exchange messages between client and server, while the RTSP and RTP over UDP protocols were used to transmit video data. A client application could request a video and specify its resolution and bitrate. VLC (VideoLAN Client) is an open-source software program that supports multiple files and streaming protocols (VLC, 2021) and is used to play the received video streams. Whereas the server streamed videos over the network using the GStreamer RTSP server. GStreamer is a GNU LGPL-licensed software library and application that allows the reading, converting, recording, editing, and streaming of audio and video files (GStreamer, 2020).

2. **Policy Manager:** is responsible for defining and reflecting policy rules to the controller and the queue manager. A policy should define QoS requirements for video in constraints that will be enforced on the network. Hence, to ensure the selection of path adheres to video streaming QoS policy constraints, SD and HD video QoS parameters were analyzed. Moreover, this thesis defined these policy rules:
    A. As thresholds for HD and SD video streams, packet loss rate and bandwidth were used as QoS parameters.
    B. For different traffic types, this work developed three types of service categorization: the first group needs quality of service requirements to be met (called group A); the

second group can accept acceptable performance guarantees (called group B); the third group does not require any QoS guarantees (called Best-Effort);

3. **Queue Manager** allows the configuration of queues and ports, in addition to other characteristics.

4.2.2 Control Plane

1. **Topology Manager** keeps track of the network topology graph, it requests and receives information from the data plane about the connected forwarding devices, new attached elements, or failed links.
2. **The Statistics Collector** collects information from OpenFlow switches and periodically polls it so the controller can get an idea of the network's state, such as the availability of resources and whether the network is congested or not.
3. **QoS Routing Manager module** is responsible for QoS-based route calculation; it applies the proposed routing algorithms to obtain a path for the flows based on its requirements. Further, it is responsible for flow admission control, determining whether the specific route is available to maintain the QoS guarantee of ongoing traffic, and informing the controller of this information. For storing calculated paths, this module uses a route cache structure. Further, the module keeps track of which resources can be admitted and which cannot by storing certain flags.
4. QoS Resource Manager's primary role is to reserve the resources for video flow, classify the traffic, and manage flow classes and queues. Also, the module task is to set up and install flow rules for new incoming flows or update existing ones in forwarding devices.

4.2.2.1 Choice of Controller

The Floodlight controller used in this work as the control plane is an open-source SDN controller developed by a community of developers. It is a Java software with a multithreaded interface that supports OpenFlow protocols 1.0-1.5. Floodlight uses the REST API (Representational State Transfer Application Programming Interface) to exchange data with external applications over HTTP (Floodlight, 2020).

4.2.3 Data Plane

The Data plane is the network topology that enables video transmission between end devices. Its switches contain queues for receiving video streaming flow and best-effort flow. Figures 4.3 and 4.4 illustrate the network topology used in this work; they were generated by the SmartDraw software (SmartDraw, 2020) that businesses and individuals use to create organizational charts, network diagrams, and flowcharts.

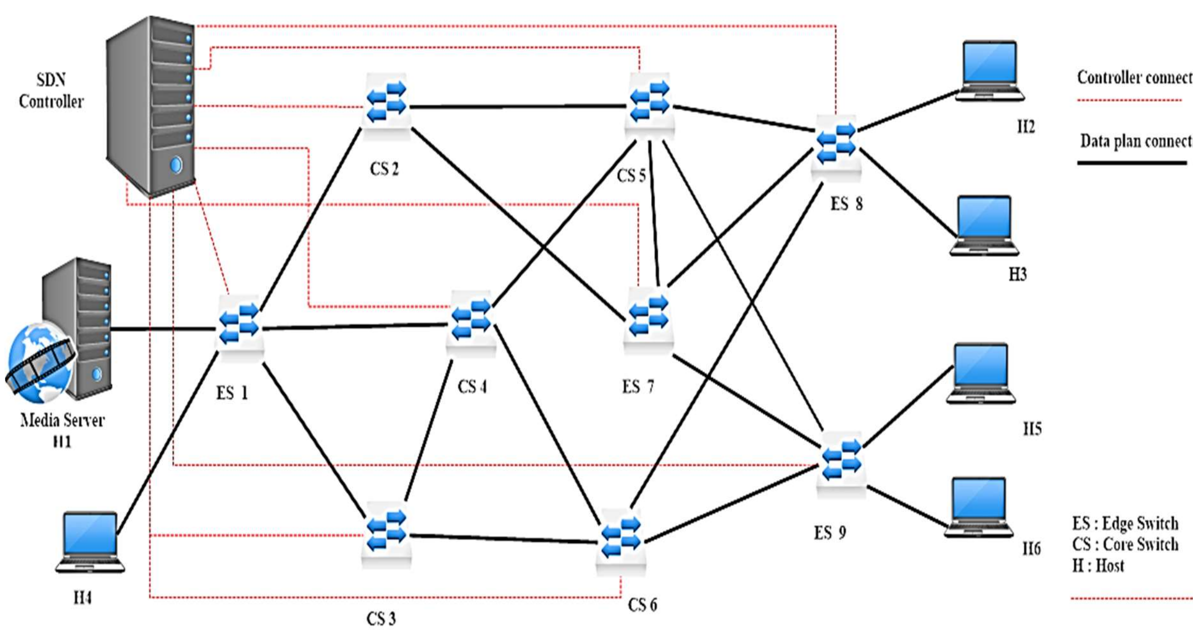

Figure 4.3: General Perspective of Topology Design

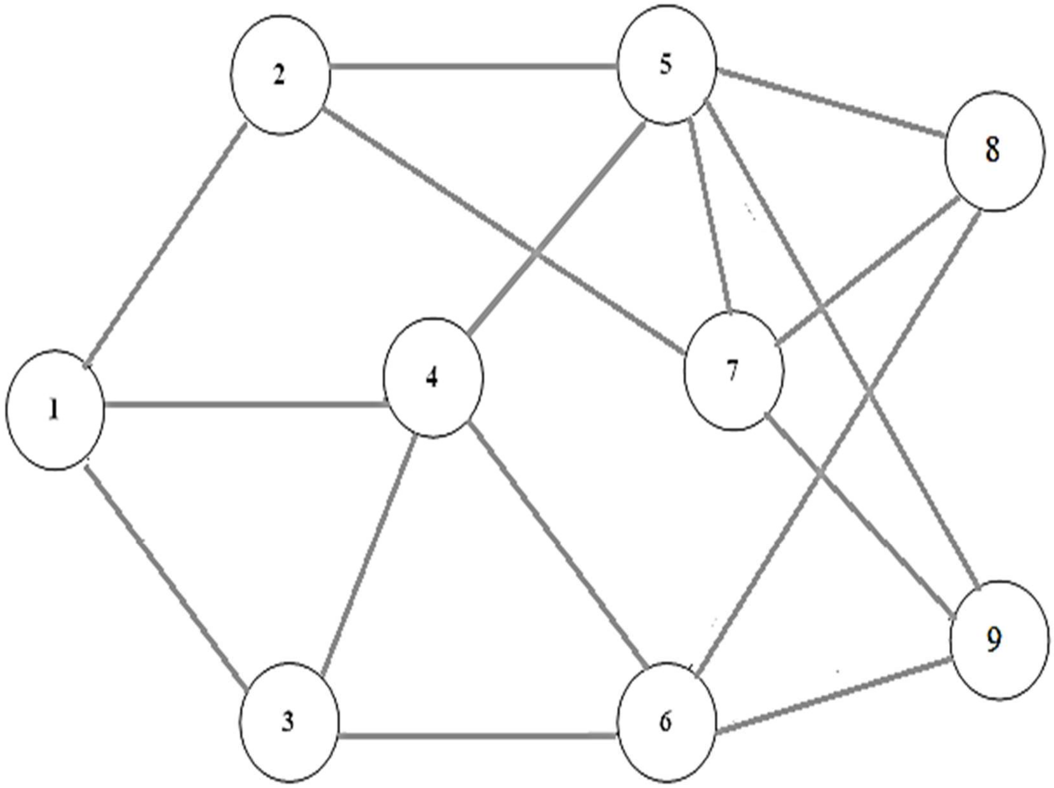

Figure 4.4: The Main Topology Graph.

4.2.3.1 Open vSwitch (OVS)

VQoSRR data plane used Open vSwitch as the network forwarding device. It is an open-source and multilayer virtual SDN switch that supports many protocols and standard management interfaces, such as OpenFlow, NetFlow, and IPV6. The original platform is Linux, and it also works on FreeBSD and NetBSD, also supported by several hypervisors and cloud computing platforms (Open-vSwitch, 2020) (Mallesh, 2017). OVS has a database (OVSDB) for storing configurations of switches. This feature enables the management of switch ports and queues across the network.

4.3 The Proposed VQoSRR QoS-Based Routing Scheme

This part describes the methodology stages followed by this research to the proposed QoS-based routing (QBR) approach as demonstrated in Figure 4.5.

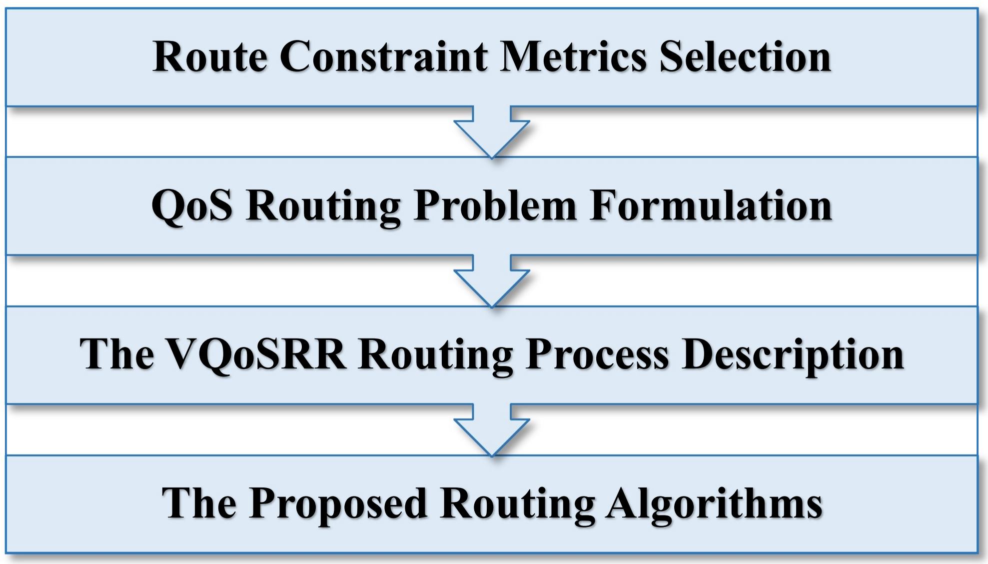

Figure 4.5: QoS Routing Methodology Stages

4.3.1 Route Constraint Metrics Selection

Video Streaming QoS requirements have to map into path metrics because the network should guarantee QoS by relying on the definition of the proper metrics. In general, finding the QBR path depends on many constraints such as additive (delay and jitter), multiplicative (packet loss), and concave (bandwidth) metrics. However, the computing complexity of choosing the best route can be very high, depending on the number of used metrics and their sequence. Wang and Crowcroft (1996) proved that finding a path subject to combinations of additive and multiplicative metrics is NP-complete if the number of metrics equals two or more metrics. Therefore, the choice of metrics and their count will depend on the importance of those metrics for the type of application (video streaming).

This thesis concentrated on enhancing the QoE of two types of video streaming resolutions, High Definition (HD) and Standard Definition (SD) video resolutions. Therefore, this work investigated the effect of different video QoS parameters that can influence QoE and focused on network-level related QoS parameters rather than video application-level parameters.

Video streaming is one of the most demanding services for link bit rates, where its demands increased with the resolution of the video content. As a result, the bandwidth is more significant for video streaming, especially when it comes to video resolution since higher video resolution means higher bitrate. Thus the network has to provide more capacity. Further, insufficient bandwidth will increase the delay and loss rate, leading to a decrease in end-user QoE. With high encoding bit rates, the stream is more exposed to packet loss, which has severe impairments on video streaming QoE, where users could experience frame freezing, complete loss of the video, or other problems depending on what video frames are lost. According to (ITU, 2008) and (Vega et al., 2018), packet loss and bandwidth have more effect on video streaming than jitter and delay. This is because the jitter could be avoided by implementing de-jitter buffers, or by the playout buffers in the video application at the receiver side. Consequently, this work applied packet loss and bandwidth as path metrics in the QoS-based routing method. Then the impact of these QoS parameters on the Quality of HD and SD video resolutions was studied.

OpenFlow port counters are used to calculate the packet loss and bandwidth on each link. The controller obtains the switch port counters by sending the “OFPortStatsRequest” message and then receives the “OFPortStatsReply”, which contains the received byte count and other statistical data.

4.3.1.1 Packet Loss as Metric

Packet loss rate (PLR) is an important measure for real-time traffic performance. These data streams must meet certain data rates for smooth transmission, hence the number of packets lost or dropped must be kept low during transmission. To obtain the PLR of the OpenFlow network, several factors should be considered, such as the OpenFlow port type if it is an output source or destination ingress port, and the TX and RX buffers where they are used to transmit and receive data. Depending on these factors, the PLR is calculated based on TX buffers within output ports and RX buffers with ingress ports, loss rate metric is associated with the specific link (i,j) between two network points, node i and node j, where $i = 1, \dots, n - 1$ and $j = i + 1, \dots, n$, and it is defined by:

$$PLR_{i,j} = \frac{\left(N_i TX\_outport \;\; - \;\; N_j RX\_inport\right)}{N_i TX_{outport}} \times 100 \tag{4.1}$$

Where N is the number of ports, RX_inport and $TX_{outport}$ are the set of buffers at the ingress port and output port interface.

Then the path loss probability between node s and node t is approximated as:

$$PLR(r_{s,t}) = 1 - ((1 - PLR_{s,i}) \times (1 - PLR_{i,j}) \times (1 - PLR_{j,k}) \times \dots \times (1 - PLR_{l,t}))$$

$$PLR(r_{s,t}) \approx PLR_{s,i} + PLR_{i,j} + PLR_{j,k} + \dots + PLR_{l,t} \tag{4.2}$$

4.3.1.2 Available Bandwidth as Metric

In order to compute available bandwidth metric through an end-to-end path between a source and destination, this work needs to find the bandwidth utilization $\mathcal{BU}_{i,j}$ and available bandwidth $\mathcal{AB}_{i,j}$ for each link (i,j) between forwarding devices i and j on the topology. Also, this study assumed that the maximum capacity of every link in network topology $C_{i,j}$ is known, and the controller will estimate the link utilization after gathering network topology statistics. So to find bandwidth utilization, suppose the OpenFlow controller received the OFPortStatsReply message at a time (t_1) contains the received bytes (RB_{t_1}), after a period of interval (ΔT) a second OFPortStatsReply received at the time (t_2) with received bytes (RB_{t_2}), the $\mathcal{BU}_{i,j}$ measured by bit/s calculated as follows:

$$\mathcal{BU}_{i,j} = \frac{RB_{t_2} - RB_{t_1}}{\Delta T} \times 8 \tag{4.3}$$

Then the available bandwidth for the link (i, j) is computed as:

$$\mathcal{AB}_{i,j} = C_{i,j} - \mathcal{BU}_{i,j} \tag{4.4}$$

To sum up, the minimum unoccupied link bandwidth among all links (i, j) along that path determines the end-to-end path available bandwidth $\mathcal{A}(r)$. This link is known as the bottleneck, and its bandwidth is considered the maximum possible bandwidth for the path, also recognized as the width path. $\mathcal{A}(r)$ for a path between node s and node t defined by the following equation:

$$\mathcal{A}(r_{s,t}) = min\ [\mathcal{AB}_{s,i}, \mathcal{AB}_{i,j}, \mathcal{AB}_{j,k}, \dots, \mathcal{AB}_{l,t}] \tag{4.5}$$

4.3.2 QoS Routing Problem Formulation

This study modeled the SDN network as a graph $G = (V, E)$, where V is a set of nodes, and $E = \{(i, j): i, j \in V\}$ is a set of links. If there is a source and destination, $s, t \in V$, the paths between s and t are represented as a set of $\mathcal{R}(s, t)$, given two constraints: $\mathcal{BT}$ is bandwidth threshold and $P\mathcal{LT}$ is packet loss threshold. The path or route metrics are the packet loss rate ($PLR(r_{s,t})$) and the available bandwidth ($\mathcal{A}(r_{s,t})$), then the QoS routing problem is to find feasible path $r \in \mathcal{R}$ joined s and t, where:

$$\mathcal{A}(r_{s,t}) \geq \mathcal{BT} \text{ and } PLR(r_{s,t}) \leq P\mathcal{LT} \tag{4.6}$$

4.3.3 The VQoSRR Routing Process Description

By using terminologies defined in the previous sections, this research proposes a flow-based routing strategy for video streaming traffic subject to meeting various QoS requirements. The primary goal of the routing strategy described below emphasizes meeting two routing constraints and efficiently determining two feasible paths for each video streaming flow in the given current state of the network. Figure 4.6 shows the general routing paradigms for finding QoS feasible paths, followed by a detailed explanation of these strategies below.

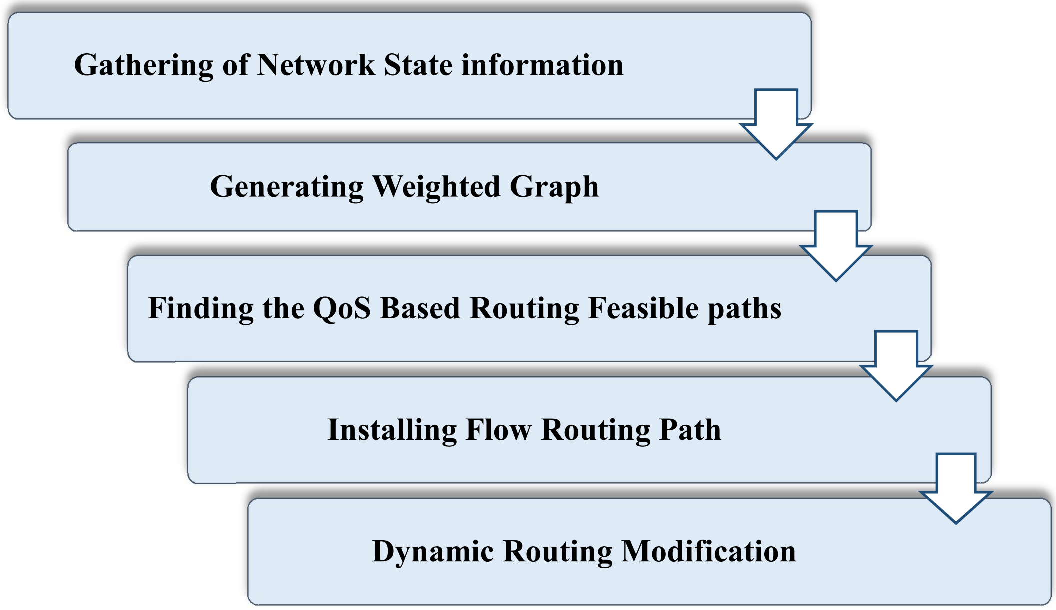

Figure 4.6: Video QoS Routing Paradigms

1. Initially, the controller uses the topology manager and statistics collector to discover the network topology and collect network status information from the forwarding elements. Then generate a weighted graph where each link is associated with packet loss rate and available bandwidth values. In these steps, the controller collects network status information from the forwarding elements. It collects OpenFlow port counters by sending the "OFPortStatsRequest" message and then receives the "OFPortStatsReply". Then the controller generates a weighted graph where each link is associated with packet loss rate $PLR_{i,j}$, and available bandwidth $\mathcal{AB}_{i,j}$. These two steps run periodically for a specified configured time. Crawley et al. (Crawley et al., 1998) stated that if the QoS metrics change frequently, this will lead to frequent routing updates, which means more computation and storage overhead. Thus the collecting link metrics should not be reliant on too much dynamism. Because of this, the study investigated the period of link status information gathering to minimize the overhead on the controller.

2. Secondly, when the server initiates a new video stream flow, the switch sends a copy of the first packet of the flow to the controller QoS routing manager to find the routing path. Then the controller acquire the link metrics $PLR_{i,j}$ and $\mathcal{AB}_{\mathrm{i,j}}$ calculated by the previous step. Then Obtain threshold constraints $\mathcal{BT}$and $P\mathcal{LT}$ from policy manager. Then obtain current network traffic service group from policy manager.
3. Next, finding QoS-based routing feasible paths by QoS routing manager algorithms if possible, where the VQoSRR determines two routing paths to balance between frequent dynamic updating of network state and reduces routing computation overheads. The idea is to use one route for the current flow routing and store the other as an alternative path for rerouting purposes. Using the alternative path presented the following advantages: 1) decreasing the time spent recalculating the routing path by utilizing the alternative route rather than using the routing algorithm again, 2) increasing response times for installing flow, and 3) providing two paths could increase resilience when a path fails.
    - Furthermore, finding the best path for multiple metrics may not exist at all. As a result, metrics precedence and sequential filtering must be defined to reduce computing complexity and find the best path (Wang and Crowcroft, 1996) (Crawley et al., 1998). Consequently, this work proposed algorithms to eliminate a subset of paths based on the available bandwidth metric. Then determine the two optimal feasible routes based on the packet loss metric. If there is no feasible route, the widest path is used as an acceptable video streaming route if the QoS policy allows it. In addition, this work uses the shortest path's algorithm to identify the best-effort path.

4. Obtain flow routing path calculated by the previous step from storage structure. Next, the QoS resource manager sends back the forwarding rules of the flow routing path to the switches by using the OpenFlow protocol (McKeown et al., 2008).
5. Finally, a dynamic routing modification happens whenever the state changes. As metrics need to be updated frequently, flow path procedures should minimize computation overhead associated with routing. Therefore, when the VQoSRR controller receives a new network status, it does not calculate a new path directly for running flows; it instead uses an algorithm and predefined flags to determine whether to use the alternative routing paths or generate a new one. These flags facilitate the admission control process defined in QoS policies. Table 4.1 shows an example of the routing path and flags storage structure. For example, the first row indicates there is ongoing flow with ID 1, and there are two paths that meet their thresholds, the first path is the current flow path, and the second path is an alternative in case the first path violates the flow thresholds. The Path count field indicates the number of paths attached to this flow; (there are two paths available, one primary and one alternative). Finally, the Admission field tells whether this flow has admitted its QoS requirement or been rejected (it admitted for Flow 1).

Table 4.1: Example of the flow path cache and flags

Flow id	**First path**	**Alternative path**	**Path type**	**Path count**	**Flow Admission Status**
Flow 1	S1-S3-S5	S1-S2-S4-S5	meets two metrics	2	Admitted
Flow 2	-	-	-	-	Rejected
Flow 3	S1-S2-S4-S7	null	meets one metrics	1	Admitted
Notes :- (S) Stand for Switch					

4.3.4 The Proposed Routing Algorithms

Next, we present the design of three routing algorithms in order to solve the QoS-based routing problem and achieve the optimization goal. There are two main ideas behind the design:

- The optimization criteria for the video streaming in this dissertation is to minimize packet loss by choosing a path with lower end-to-end packet loss rate, but it cannot exceed a specified threshold. Plus, maximize the bandwidth by choosing a path with high available bandwidth that meets video streaming bandwidth boundaries.
- The precedence between metrics is an important factor because it reflects the influence of network QoS parameters on overall QoE. Therefore, the proposed algorithm takes the bandwidth as secondary metric, and loss as primary one, since the latter is more impairing for video streaming, as it has been analyzed earlier, and there could be a route with high capacity but it can be losing the packets by different causes than congestion.

This work integrates three algorithms in order to provide QoS-aware video routing. Algorithm 1: Two Lowest Loss Widest Paths (TwoLLWPs), which determines two feasible paths between the server and the client based on packet loss rate and available bandwidth; these metrics meet the video QoE requested constraints. Algorithm 2 for adaptive re-routing. Algorithm 3 for setting up and updating flow tables by paths rules. The following sections provide the detailed descriptions of these algorithms.

4.3.4.1 Two Lowest loss -Widest Paths Algorithm (TwoLLWPs)

As discussed earlier, solving QoS problem depends on using multiple metrics known to be NP-complete. This work proposed a heuristic algorithm based on the source routing algorithm presented by Wang et al. (Wang and Crowcroft, 1996). It is considered in this work because of two features. Firstly, it computes forwarding paths on demand per flow basis, which is very appropriate for video streaming conditions. Secondly, in order to install the routing path, the entire network topology must be recognized, and this characteristic is offered by the SDN controller where it can access the full routing information of each link needed for the path computation. This research also includes in the algorithm the packet loss rather than delay, further finding two feasible paths based on k-short paths (KSP) proposed by (Yen, 1970) rather than only Dijkstra (Dijkstra, 1959). In addition, it can find the widest path based on the bandwidth metric only; if the type of QoS policy permits it. In this regard, we have defined a policy rule to determine which traffic receives the QoS. It states that the network traffic must belong to one of three service groups: restricted QoS constraints called (group A), tolerable or soft QoS constraints (group B), and best-effort (group C). The flows in group A need enforcement of their QoS constraints, and those in group B are tolerant of acceptable performance guarantees, while group C does not claim any QoS guarantees. This heuristic proposed algorithm is the main algorithm in this work (Algorithm 4.1), it is called Two Lowest Loss -Widest Paths Algorithm (TwoLLWPs), and it uses the QoS network parameters packet loss rate and available bandwidth as constraint metrics and returns the best two feasible paths if possible. The TwoLLWPs works as follows:

- There are two phases, elimination and search. First, all edges on original network topology graph that do not meet the bandwidth threshold are pruned. So, any paths in the resulting graph G satisfies $\mathcal{A}(\mathrm{r}) \geq \mathcal{BT}$. Second, when there is more than one widest path meeting QoS bandwidth requirement, the algorithm begins to search two routing paths from the source node to the destination with the lowest packet loss rate $PLR(r) \leq P\mathcal{LT}$ if it possible based on Dijkstra algorithm and Yen's k-shortest path. The two paths must meet packet loss rate restriction. The path with lowest packet loss rate is called the lowest loss-widest-first path, while the second is called the lowest loss-widest-second alternative path.

- If at least one path is found, the flow admission status is changed to admitted, otherwise it is changed to “rejected”.
- Furthermore, the algorithm does not have to find the minimum loss rate paths to all nodes, rather it finishes either when the destination node t is permanently identified or when the packet loss rate exceeds the threshold before reaching t.
- Yen’s algorithm is used to determine the first k-shortest paths by using Dijkstra to find the shortest path between two nodes. Then it starts to determine all other k-shortest paths. The TwoLLWPs modifies Yen’s algorithm by removing the first call of Dijkstra because the first path is already computed.
- If it fails to find a feasible path meeting both, loss and bandwidth requirements, the algorithm returns the widest path based on bandwidth constraint by calling the Widest_Path_Dijkstra algorithm; if the policy rule associated with the flow allows this, because the biggest available bandwidth will be more desirable metric for video traffic rather than shortest path. The widest path is computed in one condition, if the application flows belong to group B, which has tolerable restrictions.

Figure 4.7 shows the flowchart algorithm of the process described before.

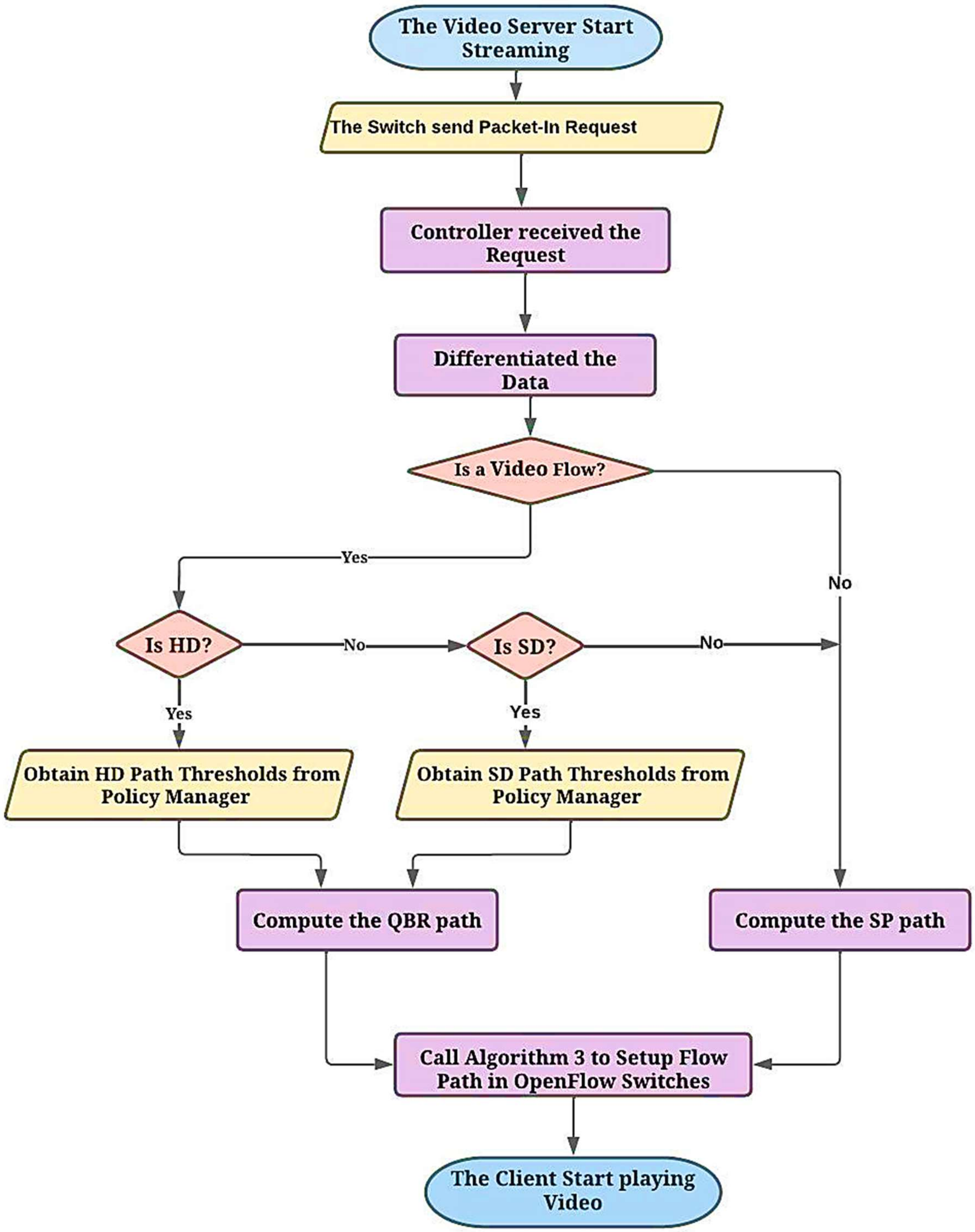

Figure 4.7: Algorithm 4.1 Flow-Chart

Besides, all the procedures mentioned above are achieved by implementing these steps in Algorithm 4.1.

Algorithm 4.1:Two Lowest Loss -Widest Paths (TwoLLWPs)

Input : Weighed $G = (V, E)$, a graph with node and edge set, each edge has two weight values the $PLR_{i,j}$ and $\mathcal{AB}_{i,j}$, values are non-negatives. A source $s \in V$, a destination $t \in E$. Two constraints $\mathcal{BT}$ and $\mathcal{PLT}$. Flow Group one of these values $FG : [A, B]$.

Output : HashMap with*: paths set, path type, path counts,* and *Flow Admission Status*: *FAS:* [*A* for Admitted*,* *R* for Rejected].

Step 1: To create $newG$, $\forall i, j$: $PLR_{i,j} = \infty$ if $\mathcal{AB}_{i,j} < \mathcal{BT}$. *// Prune phase*

Step 2: $SPT = \{s\}$; $\forall i \neq s$ $PLR(p_i)^* = PLR(p_{s,i})$, $parents$ *={}. //initialization*

Step 3 : Find $k \notin SPT$ so that $PLR_k = min_{i \notin SPT}\, PLR(p_i)^*$.

If $PLR_k > \mathcal{PLT}$, {//No path could be found

If $GF == B$, {

$r^* = Widest\_Path\_Dijkstra(G, s, t, \mathcal{BT})$

If $r^* \neq \emptyset$, $Build\_HashMap(r^*)$

Else $FAS = R$. }

$FAS = R$. }, ***End Algorithm***.

If $t \in SPT$, { *//At least one path is found*

FAS = A.

$r = Build\_Path(parents\ , t)$

KSP = YenKSP($newG$*, A[0]=r, k =2)// Call yen's algorithm to find second path*

If $KSPcount \geq 1$, $Build\_HashMap(KSP)$ }, ***End Algorithm.***

$SPT := SPT \cup \{k\}$.

Step 4 : $\forall i \notin SPT$: $PLR(p_i)^* := \min\,[PLR(p_i)^*, PLR_k + PLR_{k,i}]$, $parents[i] = k$.

Step 5 : Go to Step 3.

4.3.4.1.1 Analysis of TwoLLWPs Algorithm Time Complexity

The time complexity of this algorithm is dependent on its phases. The first phase prunes the graph by searching through all vertices and edges to eliminate links does not meet bandwidth constraints via replacing their packet loss rate ∞. This step is supposed to be executed once and requires $O(N^2)$ run time. Despite this, the floodlight controller already has a function to generate a weighted graph, so this work modified it by inserting if condition to keep only links that fulfilled the bandwidth constrain. So the graph is created from the beginning with only desirable links. The second phase is to find two paths using Dijkstra and Yen algorithms. We implemented Dijkstra using a Fibonacci heap priority queue so it will run in $O(M + N \log N)$ time rather than $O(N^2)$, consequently Yen's algorithm makes KN invocations of Dijkstra's thus takes $O(KN(M + N \log N))$ (Mohanta, and Poddar, 2012), (Bouillet et al., 2007). The proposed algorithm set $k = 2$, as a result, the complexity of using two algorithms gives following equation:

$$Compexity = O(N(M + N \log N)) \tag{4.7}$$

4.3.4.2 The Dynamic Traffic Re-Routing Algorithm (DR-RA)

Most QoS routing methods use on-demand path computation, but it has two disadvantages. First, it delays the process of forwarding traffic. Second, it involves the execution of a path computation algorithm for each flow request, adding further overhead to the routers (controllers in case of SDN), notably when the frequency of path calculation is high. Also, if the QoS metrics change frequently, this will lead to frequent routing updates, which means more computation overhead. Thus the collecting link metrics should not be reliant on too much dynamism.

Furthermore, the complexity of the routing algorithm increased processing overhead (Crawley et al., 1998), (Masip-Bruin et al., 2006). Thus, this study proposes Algorithm 4.2: called the Dynamic Traffic Re-Routing Algorithm (DR-RA) to address these mentioned issues.

The DR-RA is responsible for updating the routes cache and rerouting traffic by using the alternative path or generating a new one. Briefly, it performs in this manner: it runs periodically starting with reading the network statistics based on a predefined interval time. Next, check if the current path violates the flow QoS requirements; then deletes its flow entries from the switch. Afterward, it examined the alternative route to see if it satisfies the flow quality requirement. If not, it calculates another path. Using the alternative path presented the following advantages: 1) decreasing the time spent recalculating the routing path by utilizing the alternative route rather than using the routing algorithm again, 2) increasing response times for installing flow, and 3) providing two paths could increase resilience when a path fails. Figure 4.8 illustrates the flowchart for the Algorithm 4.2.

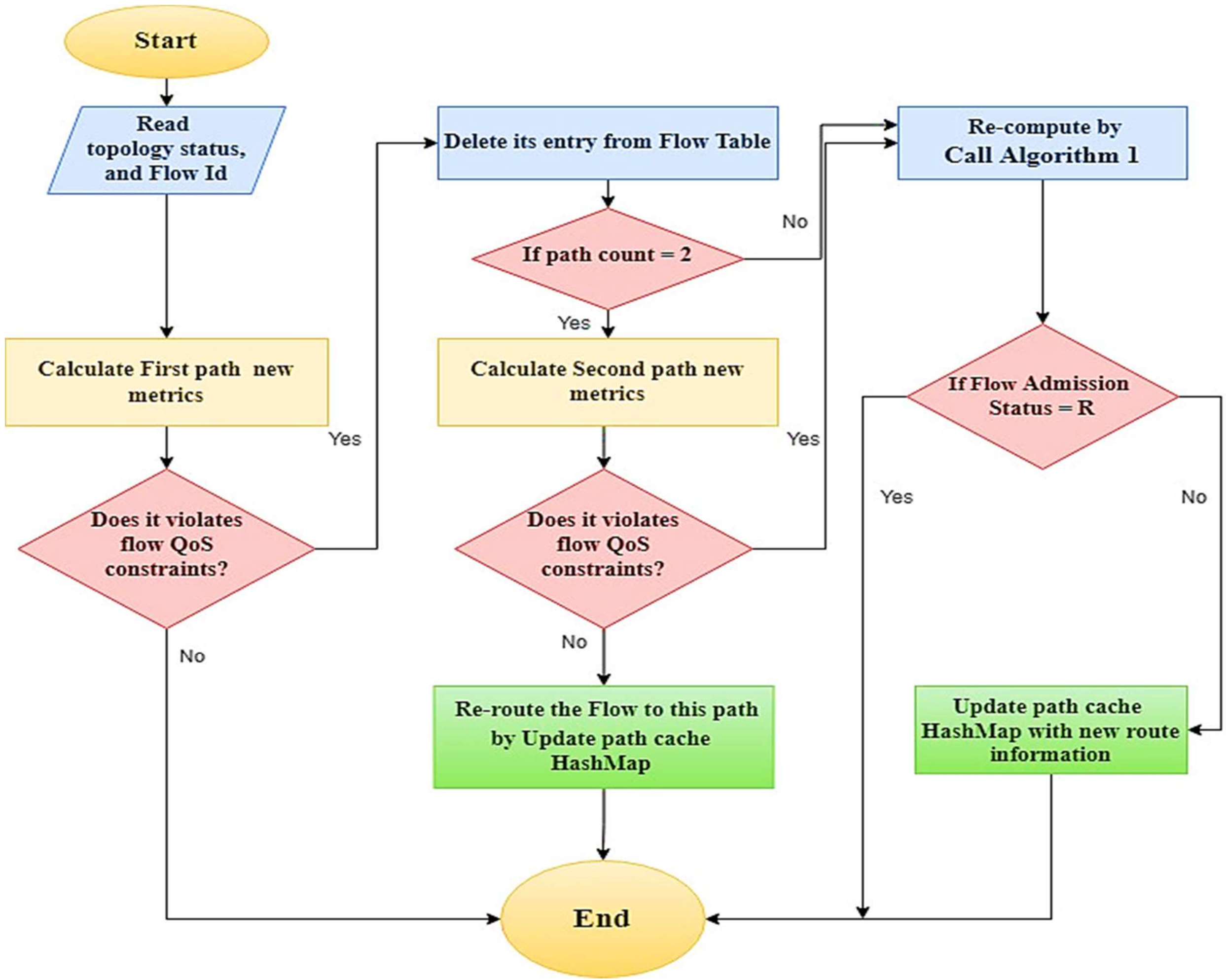

Figure 4.8: Flowchart of Algorithm 4.2 (Dynamic Re-Routing Algorithm (DR-RA)).

4.3.4.3 Installing or Updating Flow Path (IUFP) Algorithm

Algorithm 4.3 is in charge of receiving the switch packet in request and pushing the path rules into the flow tables for new incoming flow or updating tables if the configuration rules of already proceeding flow changes. In addition, the IUFP is responsible for admitting or rejecting flow requests; because the VQoSRR must route the video stream along a path that can accommodate its QoS requirements, such as meeting packet loss and bandwidth thresholds. Otherwise, it indicates that the QoS currently requested cannot be admitted. Next, the Algorithm 4.3 that illustrates this scenario.

Algorithm 4.3 : Installing or Updating Flow Path (IUFP)

Input	**:**	Packet_In request
Output	**:**	Packet_Out response

Step 1	**:**	//Identify if the request for new or for proceeding flow by searching in the HashMap paths cache using flow Id:
		if *flow Id not found, then*
		*isNewFlow go to **Step 2**.*
		else
		*isOldFlow go to **Step 3**.*

Step 2	**:**	Call Algorithm 1 (TwoLLWPs),
		if: *Flow Admission Status == R,* then
		delete its flow entry from HashMap, reject this flow, and end the procedure.
		else
		pick the first path from HashMap, send the Flow path response to the switches, and end the procedure.

Step 3	**:**	Call Algorithm 2 (DR-RA),
		if: *Flow Admission Status == A,* then
		pick the first path from HashMap, send the Flow path response to the switches, and end the procedure
		else
		delete its flow entry from HashMap, reject this flow, and *end the procedure.*

4.4 The Proposed VQoSRR Reservation Scheme

Frequently changing traffic metrics may cause QoS-based routing to consume network resources and overburden the controller as a result of frequent routing updates. Hence, the proposed solution relied not just on QBR, but also on queuing mechanisms to achieve a balance between routing overhead and quality assurance and control of the data flow. Incorporating resource reservations with QoS routing leads to fine control over the route and resources; this allows better congestion management, thus reducing route updating and providing more stability for QoS routes. Additionally, this method also contributes to reducing latency, a factor that can impact video streaming QoE.

The VQoSRR suggested defining a higher-level reservation control strategy with suitable administrative mechanisms to enable fairness to flow according to their requirements and coupled with a method to configure the underlying resources. The idea is to utilize the rate and per-class queueing reservation mechanism to prioritize flows. The task is to map application traffic to different QoS levels, each assigned a rated weight then partitioned queues based on the level specified rate weight. Afterward, marked each packet to a class with the Differentiated Service Code Point (DSCP), and according to that, the network forwarders handled packets.

Figure 4.9 shows the main components of the methodology followed by this study to allocate resources for QoS flows and non-QoS flows. Further explanation is provided in the following sections.

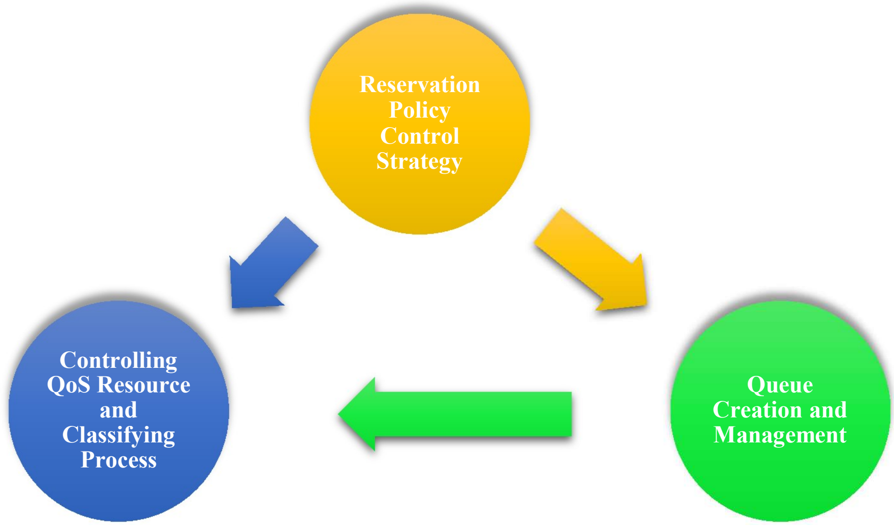

Figure 4.9: The Main Components of Resource Reservation Methodology.

4.4.1 Reservation Policy Control Strategy

The proposed policy mechanism calculates a weighted rate for each expected flow, then a portion of the total link rate is assigned based on this weight. The study considers employing the QoS thresholds and application service category (strict or soft constraints) as parameters to estimate the weight. The proposed policy defines QoS thresholds as ranges at the policy manager, then different applications flows are grouped to one of these ranges by the administrator. Consequently, let FG_x be aggregated flow group under a specific threshold range ($\mathcal{TR}p_j$), where $1 \leq j \leq n$ and p indicates the precedence of the $\mathcal{TR}$ according to its FG quality requirements, whether they are high or low, so that the highest requirements take the first precedence, followed by the next highest, and so on.

Moreover, assume FG_x belongs to one of three service categories: strict or hard QoS constraints, soft or tolerated QoS constraints, and best effort category without any guarantees. So let $\mathcal{S} \in \{k, l, m\}, k > l > m,$ where $\mathcal{S}$ is a parameter denoted to the service category. Also, assume that each threshold range and flow group take the degree of importance to let:

- $\mathcal{Y}$ importance factor of $\mathcal{TR}$ of FG_x.
- $\mathcal{Z}$ importance factor of FG_x.

Based on the above assumptions, the objective is to calculate a weighted bandwidth rate ($\mathcal{w}r$) for FG_x from the total link rate $\mathcal{R}$in according to the importance of their QoS constraints. Firstly, if the threshold range precedence is smaller than all other threshold ranges then the importance factor ($\mathcal{Y}$) will be higher, which is formulated as follows:

$$\begin{gathered} \text{if } \min \left[\mathcal{TR}p_j{}^{FG_1}, \mathcal{TR}p_j{}^{FG_2}, \mathcal{TR}p_j{}^{FG_3}, \dots, \mathcal{TR}p_j{}^{FG_n}\right] \\ = \mathcal{TR}p_j{}^{FG_x} \ \forall\, x \in i, 1 \leq i \leq n \\ \textit{then let} \\ \mathcal{Y}^{FG_x} = \mathcal{N} \\ \textit{where } f_x(\mathcal{N}) > f_{i-x}(\mathcal{N}) \end{gathered} \tag{4.8}$$

where $\mathcal{N}$ is a number representing a higher weight.

Secondly, from (4.8), the importance of the flow group is determined from the threshold range importance and service category as,

$$\mathcal{Z}^{FG_x} = \mathcal{Y}^{FG_x} + \mathcal{S}_i \tag{4.9}$$

Consequently, multiple service classes are created, one class for each flow group $(FG_x, \mathcal{C}_x)$. Then each class of service $\mathcal{C}_x$ assigned queue with a given wr from $\mathcal{R}$ where wr is found as

$$wr_{\mathcal{C}_x} = \frac{\mathcal{R} * \mathcal{Z}^{FG_x}}{100} \tag{4.10}$$

Where,

$$\sum_{i=1}^{n} wr_{\mathcal{C}_i} \leq \mathcal{R} \tag{4.11}$$

4.4.2 Queue Creation and Management

Babiarz et al. (Babiarz et al.,2006) state that multimedia streaming services need to use a rate queueing system where schedulers set a minimum, a maximum, or both rates. Therefore, this study allocates the bandwidth for each flow group using rate limiting, through using a queue manager. It operates as a standalone application, developed in shell script combined with Python to accomplish queue configurations at network switches; this includes the creation of queues, prioritizing them, assigning bandwidth rates, and destroying the configuration settings. The application communicates with the policy manager to obtain the set-up parameters, including the number of classes and the weighted rate for each class.

4.4.3 Controlling QoS Resource and Classifying Process

Whenever a video server initiates a new stream, the SDN controller QoS resource manager receives the first packet as a packet-in request; after that, it creates a flow entry on the edge switch connected to the server, enabling the high-priority flow with the DSCP field. The alteration of packet fields must be done at the border of the network, as forwarding decisions are made based on the DSCP value. Then regular DiffServ forwarding could take place inside the core network. The controller classifies any arriving flow by application traffic transmission protocol, source port, and destination port. Traffic classification is performed corresponding to the QoS policy settings. After that, the controller creates match and action rules at switches to direct the flow to its specified queue based on its DSCP. Figure 4.10 gives an exemplified explanation of how the resource reservation method works.

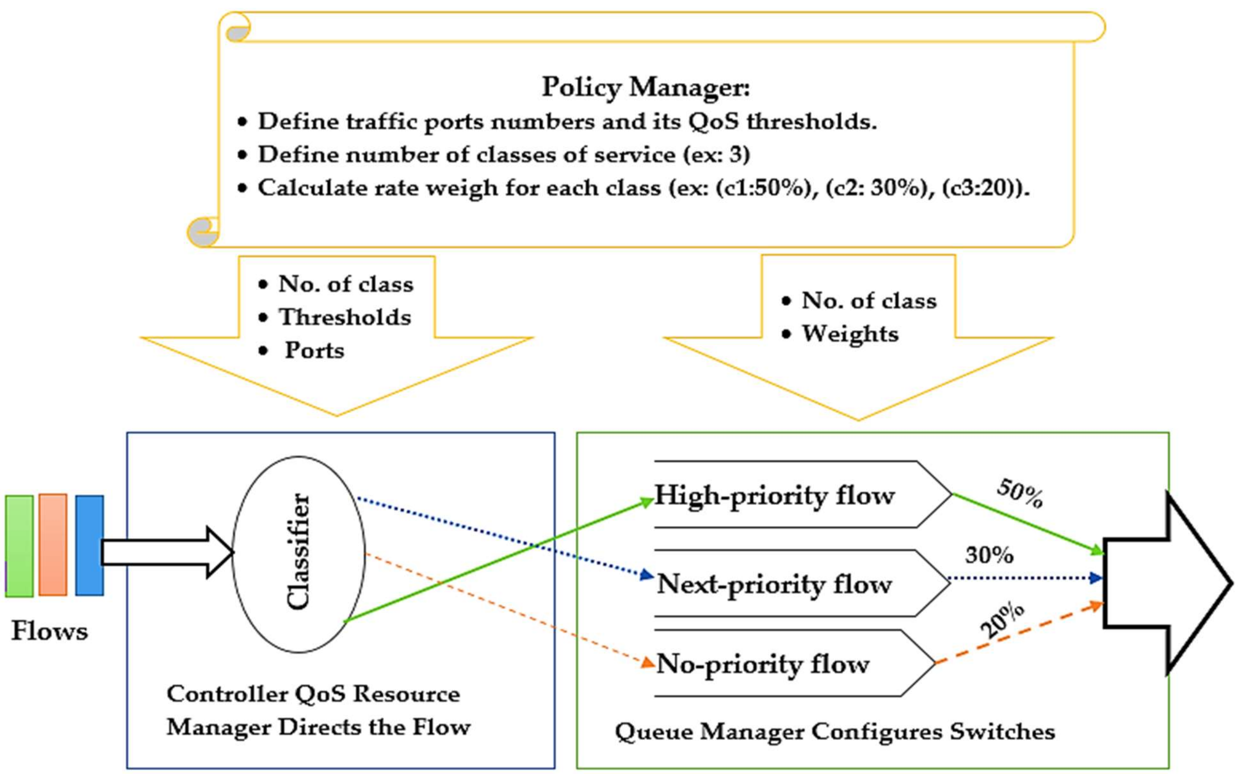

Figure 4.10: Illustrative Example of the Queues Allocation Process.

4.5 Evaluation and Validation

To validate the proposed methodology, several metrics have been developed for evaluating the QoS/QoE of video. These evaluation metrics are introduced in this section in terms of video streaming services, and network QoS. In addition, this section also presents the measurement and simulation tools used in this study.

4.5.1 QoE and QoS Measurement Metrics

Generally, video streaming QoE can be measured by subjective or objective methods or both of them. The proposed model was validated using both subjective and objective metrics of QoE. Also, this study used the network throughput to reflect the enhancement of network QoS.

4.5.1.1 Subjective Method

The subjective method is the most accurate technique for measuring perceived video quality because it reflects the perceptions of users who use the service. It is conducted by asking subjects to rate the video they have been watching, the subjects rate the quality as excellent, good, fair, poor, or bad. The Mean Opinion Score (MOS) (P. ITU-T, 2008) includes well-known QoE scales that represent an average of scores across subjects. MOS maps ratings between excellent and bad to numerical values between 5 to 1. This thesis followed a methodology consisting of conducting subjective tests with real end-user participants in order to evaluate video quality, the quality of the videos rated in this work is based on the MOS metrics. A total of 15 viewers rated three different video quality test cases. Each user was asked to watch video sequences of the Caminandes Llamigos videos (with resolutions of 1920 x 1080 and 854 × 480). All test video samples were played in the VLC media player. The testing was conducted on an HP laptop computer, with Intel core™ i7-7500U, 2.70 GHz CPU, 2 Cores, and 16 GB RAM. This laptop comes with 15.6-inch screen size and AMD Radeon R7 M440 dedicated graphics card. Subjective tests were conducted in a separate room without outside interference. Subjects received a paper copy of the questionnaire before the tests.

4.5.1.2 Objective Method

Objective methods are used to measure QoE based on objectively measured network and video parameters in the absence of the human viewer. The full reference (FR) model is one of the objective estimation methods for video streaming QoE. It works by accessing both the original, called a reference, and processed video, called distorted, to assess the quality. The properties of the two tested videos are compared frame-by-frame to check different features such as contrast features, and color processing, etc. There are two types of FR measurement objective metrics used in this study: The Structural Similarity Index Metric (SSIM) (Wang et al., 2004) and the Video Multimethod Assessment Fusion (VMAF) which has been recently proposed by Netflix (Li et al., 2016).

SSIM measures the video quality of streaming video on receiving devices. SSIM scores range from 0 to 1. It measures the similarity between the reference video and the streaming output video. A score of 1 indicates that the output video is identical to the reference video. A lower score represents a greater loss in quality compared to a reference video. The standard formula for Structural Similarity Index between frames x and y can be defined as (Wang et al., 2004):

$$SSIM(x,y) = \frac{(2\,\mu_x\,\mu_y + C_1)(2\,\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)} \tag{4.12}$$

Where x and y are the two images being compared. μ_x is the pixel sample mean of x, μ_y is the pixel sample mean of y. C_1 and C_2two are variables to stabilize the division with weak denominator. σ_{xy} is the covariance of x and y, and σ_x^2 the variance of x, and σ_y^2 the variance of y.

In VMAF, quality scores range from 0 to 100 per video frame, 100 being the same quality as the reference, while 0 being the worst quality.

4.5.1.3 Network Throughput

Network throughput refers to the rate of successful message transmission over a communication channel. It is measured in bits per second (bit/s). The reason for studying throughput is to evaluate the improvement in the quality of service provided by the proposed model. Throughput was studied to reflect the impact of packet loss with and without the proposed VQoSRR.

4.5.2 Measurement and Simulation Tools

To measure the quality of service, the delivered video on each client was recorded and compared with the original video sequence using the MSU Video Quality Measurement tool (MSU, 2021). The software is used to evaluate the objective quality of the video. It supports both full and single references. Most objective video quality metrics are supported, including PSNR, Structural Similarity (SSIM), Mean Square Error (MSE), Video Quality Model (VQM), and VMAF. To measure video quality, the tool requires both the original and the delivered video.

The environment software used to simulate the proposed VQoSRR network topology is Mininet (Lantz et al., 2010). This emulator creates networks, switches, controllers, etc., it runs on standard Linux network software and supports OpenFlow switches and Software-Defined Network. The controllers will be deployed remotely outside Mininet. While the data plane was deployed using Open vSwitch (OVS) inside Mininet.

CHAPTER V

INVESTIGATED EXPERIMENTS RESULTS and ANALYSIS

5.1 Introduction

This chapter presents the experimental assessment undertaken to evaluate the proposed VQoSRR methodology. The simulation testbed environment and parameters are described. Additionally, different scenarios were designed to study the performance of the proposed solution. Then, evaluation results are analyzed and discussed. Three experiments were accomplished to test the performance of VQoSRR in terms of studying the two lowest losses-widest paths algorithm (TwoLLWPs), the dynamic re-routing algorithm, and the reservation method. Next, sections describe the details of each experiment.

5.2 Testbed Setup

The simulation was performed in the Mininet emulator and implemented the network topology described in chapter 4. Tables 5.1 and 5.2 illustrate the network parameters plus software and hardware specifications.

Table 5.1: General Network Parameters

Topology Parameters	**Values**		
Network links packet loss rates (%)	Ranging between: 0.0, 0.001, 0.005, 0.05,0.5, 1.0, and 2.0		
Link bandwidth	Ranging between: 100 Mb/s, 50 Mb/s and 10Mb/s		
Number of links	22		
Number of Nodes	1 Controller	9 switch	6 hosts one of them is Server

Table 5.2: Used Hardware and Software

Hardware or Software	Specifications
Ethernet Link Bandwidth	100 Mb/s
Computer OS Specifications	Microsoft Windows 10 Home, x64-based PC System and Linux Ubuntu 18.04.5 LTS (Bionic Beaver) OS, x64-based PC System
Programming Languages and Editors	• Java JDK 8. • Python v3.8. • Eclipse Editor. • PyCharm Editor.
Video System and Measurement Tools:	• GStreamer v1.18.4. • VLC Player. • Wireshark packet analyzer. • MSU Video Quality Measurement Tool 12.0 beta.

5.3 TwoLLWPs Algorithm Experimental Results

The TwoLLWPs algorithm was tested in three test cases to illustrate the importance of each QoS parameter and its impact on QoE. The first test case compared the floodlight default mode (FDM) with video QoS mode (VQM). When using FDM, all data traffic was routed by the original floodlight shortest path algorithm. In contrast, VQM transmits two types of data traffic, videos (combining two types, SD and HD) and best-effort. The video is routed through the generated paths by the TwoLLWPs algorithm, while the best-effort data remain at floodlight shortest paths. The second test case studies the various packet loss rate constraints that influence the video streaming QoE. Finally, the third test case examines the impact of different bandwidth constraints.

5.3.1 FDM vs VQM Test Case

The SDN network link bandwidth used in this test is 100 Mb/s, and the links of the longest paths have no loss, while other links take values> 0 and <= 2 % loss metric. The iperf testing tool generated background traffic between hosts 4 and 6 (UDP traffic at 10 Mb/s) (Iperf, 2021). The test video samples used are the Sintel Trailer, obtained from the Sintel website copyright under Blender Foundation (Sintel Trailer, 2021), and Caminandes videos under the Creative Commons Attribution 3.0 license from the caminandes.com (Caminandes, 2021). The policy rule categorizes the flow as belonging to the group (A). QoE and QoS parameters and their values are summarized in Tables 5.3 and 5.4.

Table 5.3: Video Simulated QoE Parameters

Video Name	Type	Bitrate kbps	Size MB	Duration	Frame/Second
Sintel Trailer 480	SD	540	4.16	52 s	24
Sintel Trailer 1080	HD	2116	13.9	52 s	24
Caminandes_llamigos_1080p	HD	1042	191	2:30 m	24

Table 5.4: The Network Simulated QoS Parameters

Video Type	FDM	VQM	
		Packet Loss Rates Constrain Metrics	Bandwidth Constrain Metric
Class 28 (SD)	-	0.5%	3 Mb/s
Class 30 (HD)	-	0.005%	10 Mb/s

Figure 5.1 shows the variance between using FDM and proposed video QoS-based routing. The fact of missing frames has a significant influence on FDM, which results in a bad viewing experience compared to VQM because of using the shortest path rather than QoS routing. The video artifact problems seen in FDM include annoyance and blocking in frame pixels. Also, sometimes complete image losses are shown. The frame in VQM is relative to the original frame, with few color distortions.

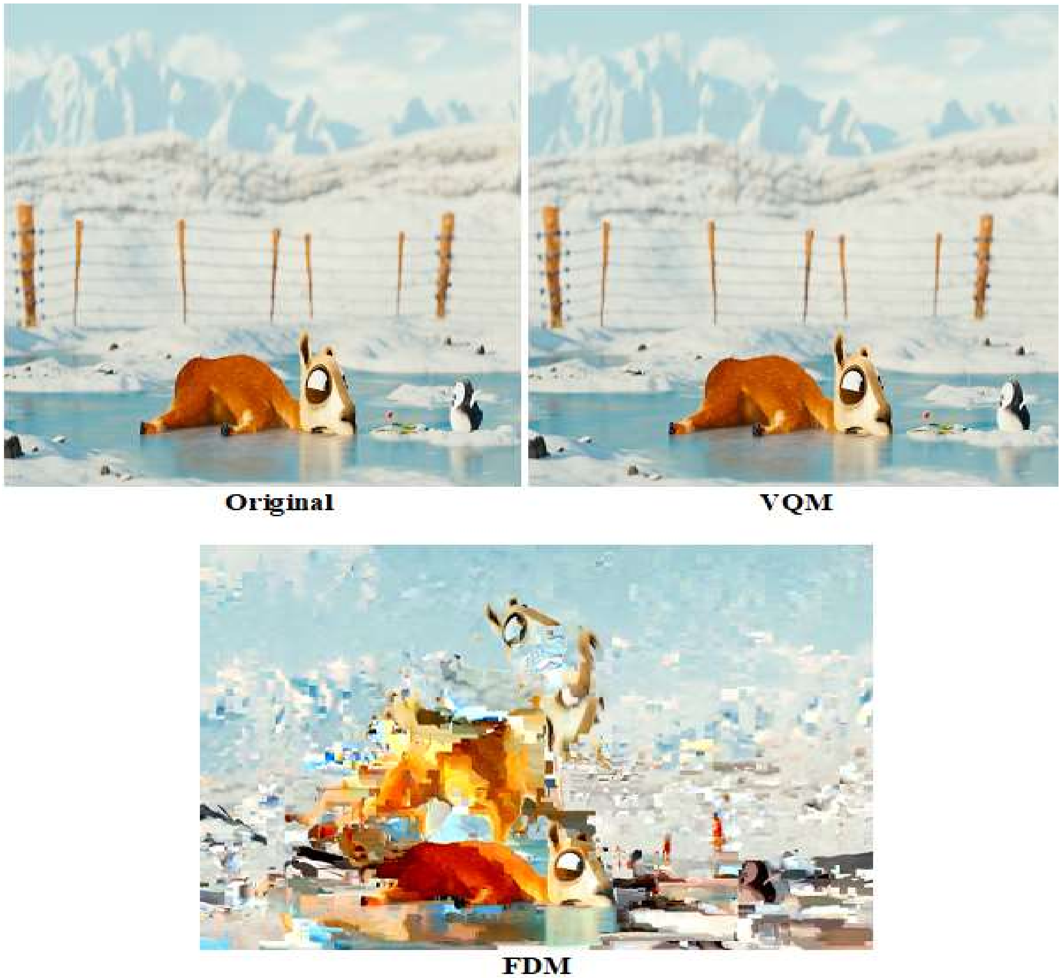

Figure 5.1: Original Video vs FDM and VQM

Figures 5.2 and 5.3 show the impact of the TwoLLWPs and FDM on different received video sequence resolutions. The X-axis represents tested frames, and the Y-axis represents the structural similarity index metric (SSIM). As observed, the video QoE is affected by QoS network parameters. In this experiment, the perceived quality of video dropped without the QoS-based routing scheme. For SD type, the obtained SSIM values obtained for the Sintel (480) video indicate higher quality viewing behavior. Additionally, HD performance results also improved when compared to shortest path routing.

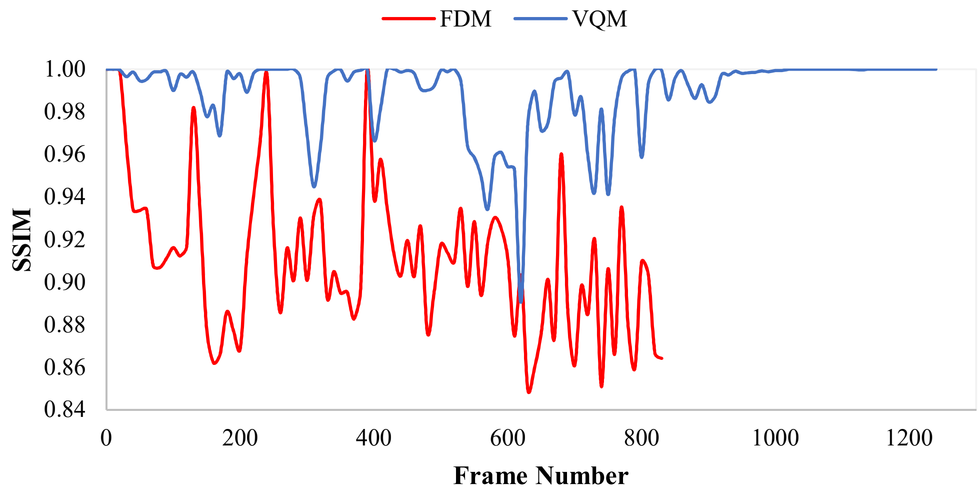

Figure 5.2: FDM vs VQM impact on SD videos

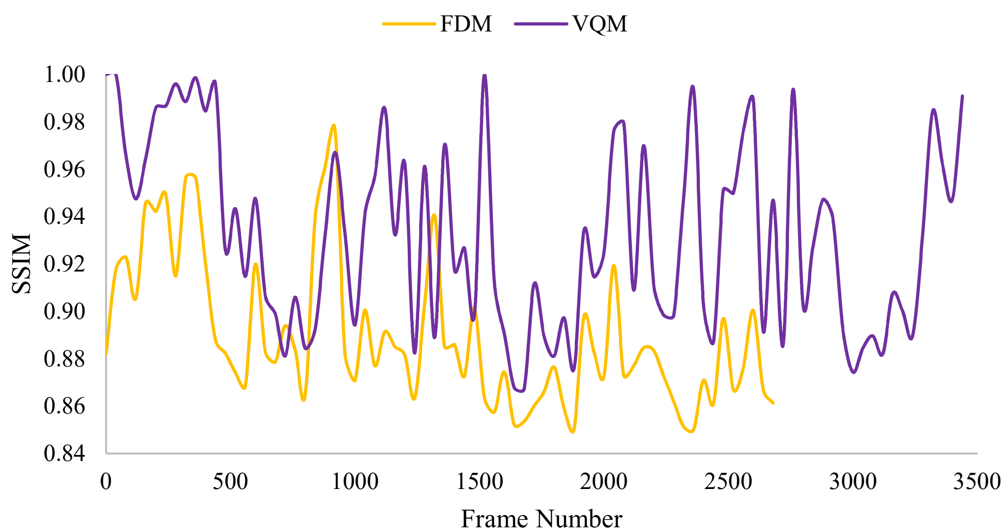

Figure 5.3: FDM vs VQM impact on HD videos

In addition, Figure 5.4 displays the mean VMAF scores for Sintel Trailer samples. The results exhibit that the proposed approach offers high scores, indicating good viewing quality for SD and fair for HD. In contrast, FDM's highest average value does not exceed 30 for SD, for HD's value is less than 10, which suggests a poor and bad perception of QoE.

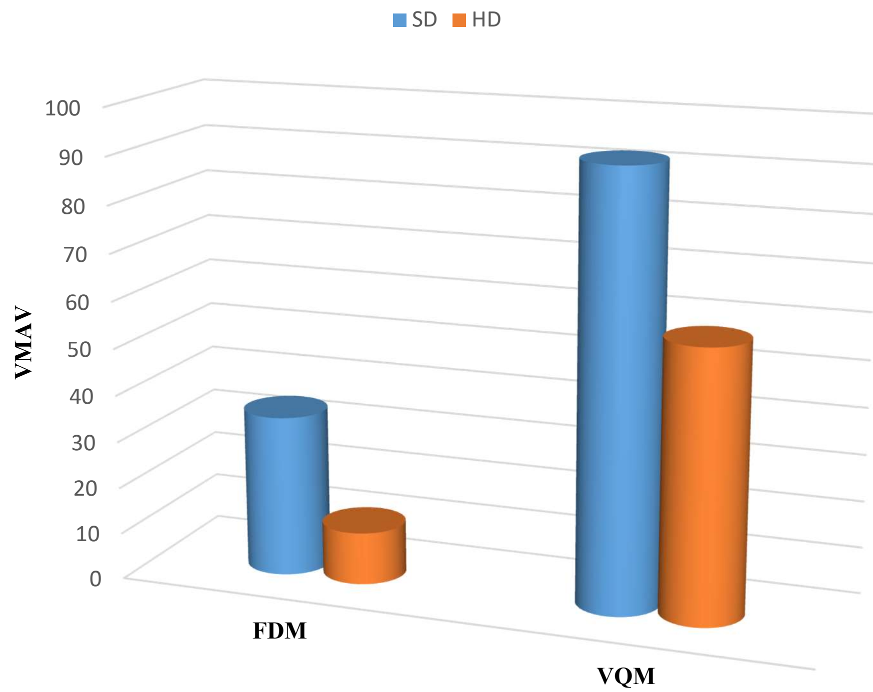

Figure 5.4: Average values of VMAF Objective Quality Metric

Also, we investigated the network throughput achieved in the case of using FDM and VQM. This time we used the Caminandes Llamigos video (with a resolution of 1920 x 1080) and Wireshark packet analyzer. Figures 5.5 presents the video traffic throughput for FDM, and Figure 5.6 shows the network throughput received for the proposed model. As noted in the first Figure 5.5, the throughput decreases significantly due to the packet loss increase, with the maximum value reaching only 2,500 bytes per second. The second Figure 5.6 shows that VQM achieves a higher throughput where the maximum throughput range between 400,000 and 500,000; the reason is that the TwoLLWPs reduce the packet loss probability and provide information on the available bandwidth, which improves the network throughput.

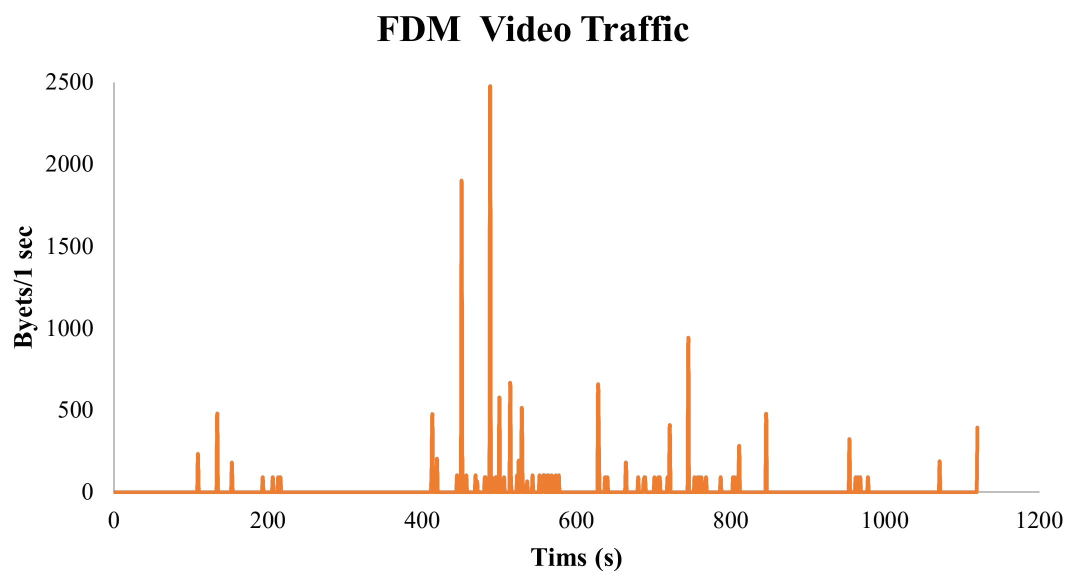

Figure 5.5: FDM Network Throughput

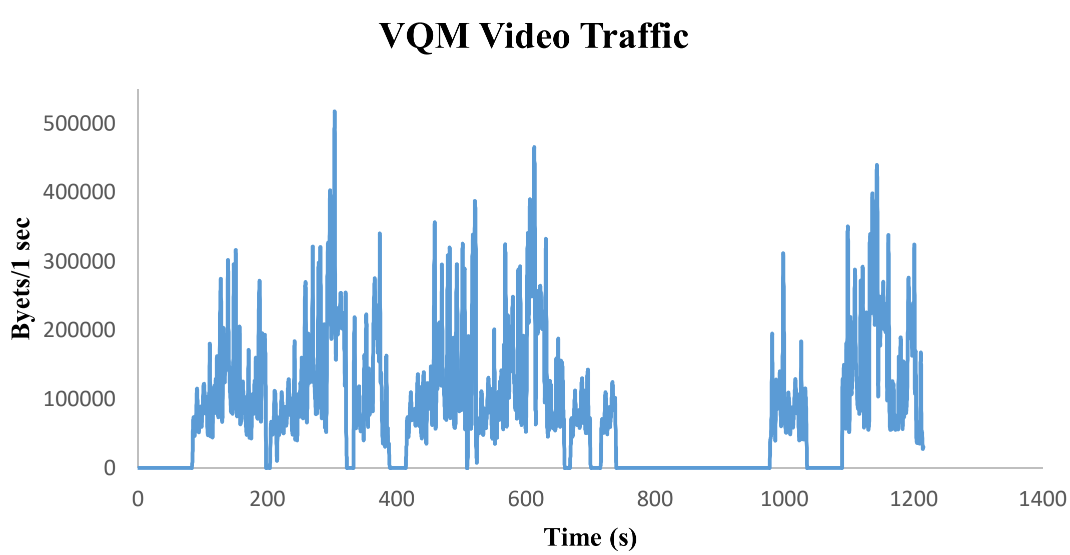

Figure 5.6: VQM Network Throughput

From the results above, this section can make many remarks. For example, the number of video frames received by VQM is higher than the number of frames received by the FDM; because the proposed QoS-based routing algorithm avoids lossy links. Moreover, HD video resolution produces lower objective SSIM and VMAF metrics ratings than SD resolution; because HD frames contain more information than SD. Therefore, in case of packet loss, the HD incurs more degradation in the video viewing experience.

5.3.2 Influence of Packet Loss Rate as Constraint

This test examines the impact of different packet loss values as constraints on the proposed routing algorithm. The parameters deployed in the experimental set-up are the following ones. The video samples used for this test were Sintel Trailer SD and HD, and the policy rule categorizes the flow as belonging to the group (B). The link capacity has been set to 10 Mb/s. The links of the longest paths are divided into two parts: (i) part with no loss, and (ii) part with 0.3% loss, while the other takes value > 0 and $<$ loss metric. The background traffic amount is set to be about 3Mb/s. The loss rates constraint metrics used for this test are 0.001, 0.005, 0.05, 0.5, 1.0, and 2.0%, while the bandwidth constraints metrics are fixed to 5 Mb/s for SD and 10 Mb/s for HD. For the FDM, the metrics settings in floodlight have no effect. Finally, the video quality was measured by SSIM.

Figure 5.7 presents a sample of received HD video by the TwoLLWPs with different loss rate metrics. FDM is compared with the original video. The results suggest that the QoE of the coded video increased when using the proposed algorithm. As noticed even with metrics up to 2% PLR the perceived impairments are less than the default shortest path.

Figure 5.7: Impact of setting the TwoLLWPs by packet loss rates (0.005% or 2%) against FDM.

Figure 5.8, Figure 5.9, and Figure 5.10 show the impact of different packet loss rate values on SD and HD videos. Figure 5.8 shows when the PLR=0.05% SSIM decreases, a similar result when PLR=1%, when PLR=0.5% video has higher SSIM values. The degradation of quality when we use small loss rate values is due to the probability of rejection by the algorithm because it takes more time to find feasible paths. If it fails to find one, it takes the widest path since the policy rule used in this test configures the flow to belong to group B, which has soft constraints, and the widest is considered acceptable for this group, however, this reduced SSIM. In contrast, higher loss rates initially result in higher scores as the probability of finding paths meeting the two constraints increases. However, the video quality significantly improves after a while when the controller reads the new network states.

In a similar manner to SD video, Figure 5.9 presents the SSIM values measured during HD video delivery, when the PLR=0.005%, the SSIM decreases at the beginning but improves later, and when PLR=0.5%, SSIM scores are higher at the start of video reception but, it dropped afterward. We explain this for the same reasons mentioned in the analysis of SD results previously. Moreover, when the packet loss rate is high, it indicates degraded received QoS, e.g., when PLR=1%, PLR=2%. Finally, Figure 5.10 shows that when the values of packet loss rate metrics are so close to each other, there is not much difference in the observed quality. Accordingly, as can be seen from the figures, choosing a loss metric that is not too small will enhance the performance of the routing algorithm; this must, however, be in balance with selecting the appropriate threshold.

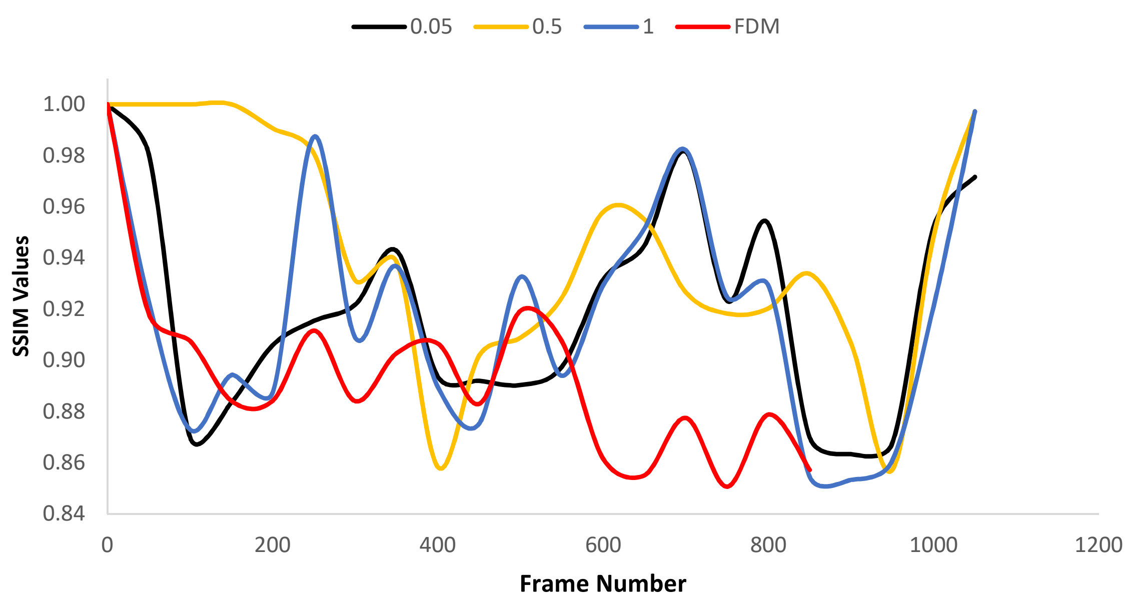

Figure 5.8: Impact of many PLR values on QoE of SD video

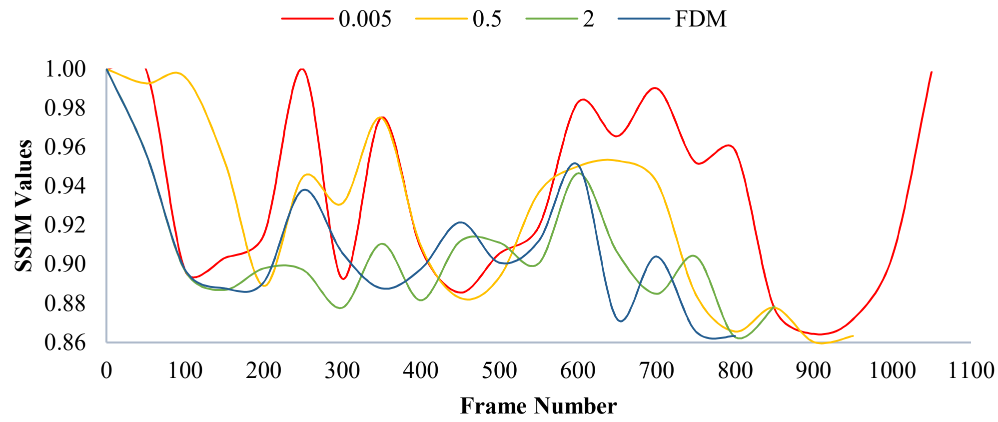

Figure 5.9: Impact of Several PLR on QoE of HD video

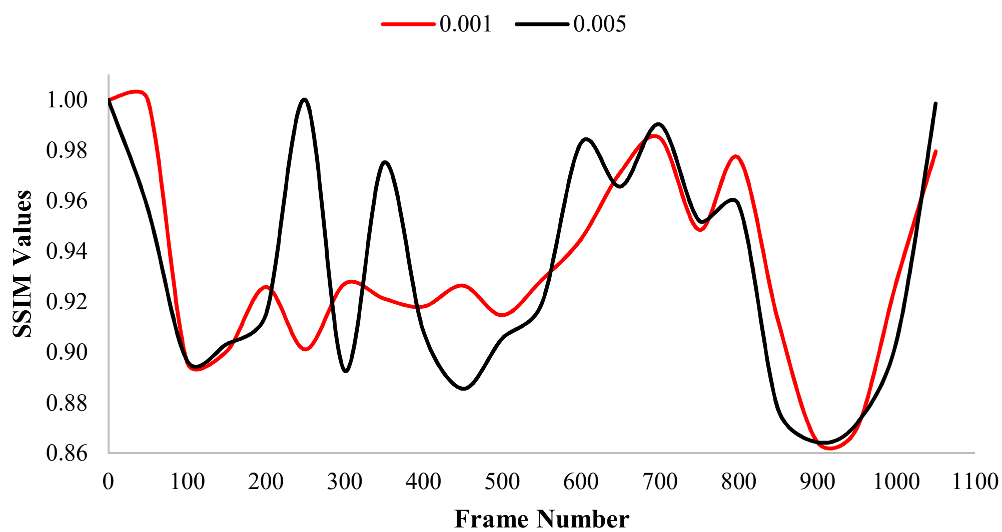

Figure 5.10: Impact of Several PLR (0.001 and 0.005) on QoE of HD video

5.3.3 Influence of Bandwidth as Constraint

In order to evaluate the bandwidth metric impact on the performance of the routing algorithm, different bandwidths have been applied (1, 5, 10, and 20 Mb/s). The link capacity is set to 50 Mb/s, and two hosts generate about 20% UDP best-effort traffic. The longest paths have no loss and the rest take values > 0 and <= 2%, and the loss input metrics are 0.5% and 0.05% for SD and HD respectively.

Figure 5.11 presents the obtained results. It shows that packet loss rate strongly affects videos more than bandwidth. As we can see, in Fig. 15, the differences in the results are irregular; e.g., in SD, the average SSIM of 10 Mb/s obtains better results than that of 20 Mb/s, despite that the 20 Mb/s means the chosen path is supposed to have a rate higher than 20 Mb/s. We can explain the behavior by the fact that the selected route ultimately depends on the loss metric; for example, the admitted path rate for the threshold=10 might be 40 Mb/s, and for the threshold=20, it might be 22 Mb/s. As well, as the same observations in the case of HD videos. Moreover, due to the best effort that competes for resources, the quality of videos may have dropped since QoS-based routing lacks a mechanism to reserve resources. Finally, small bandwidth metrics increase the options for routing paths, while high bandwidth metrics reduce these options and increase rejection rates.

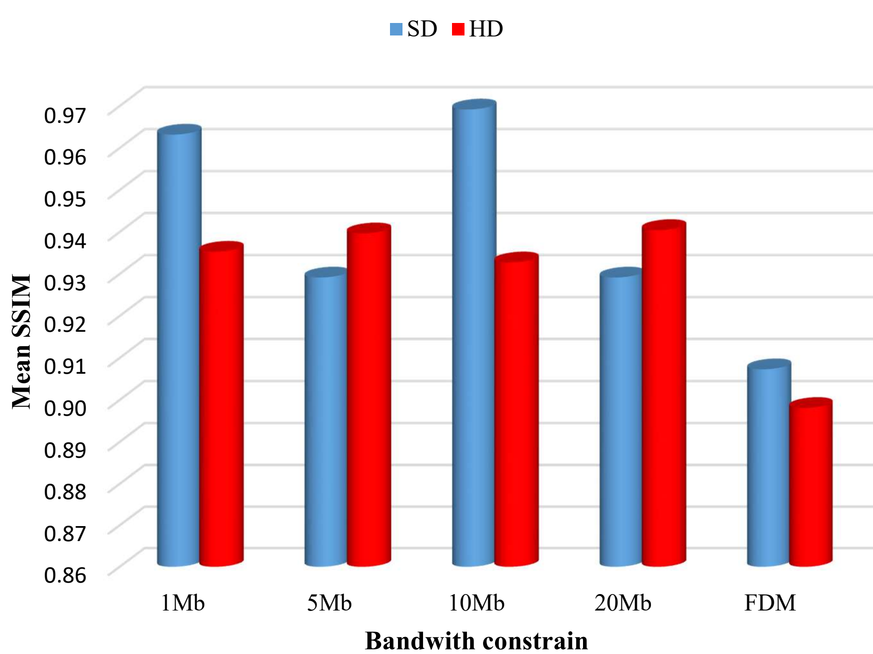

Figure 5.11: Impact of several bandwidths on QoE of SD and HD video

5.4 Dynamic Rerouting Algorithm Experimental Results

This section aims to assess the performance of the dynamic traffic re-routing algorithm (DR-RA). The simulation was carried out with the following parameters. The video specifications used in this experiment are shown in Table 5.5. The link capacity is set to 50 Mb/s. The majority of links have a loss of 1%, while links on the longest paths have no loss. There are two hosts with the best effort (UDP traffic with 5 Mb/s). For SD and HD, bandwidth constraint metrics are 3 and 10 Mbps, respectively, while packet loss metrics are 0.5% and 0.005%. The controller collected the statistics every 4, 6, or 9 seconds, whereas the DR-RA execution occurred at t= 5, 7, or 10 seconds.

Table 5.5: Videos Simulated QoE Parameters

Video Name	Type	Bitrate kbps	Size MB	Duration	Frame/Second
caminandes_llamigos_480p	SD 854 x 480	847	17.5	1:30	24
caminandes_llamigos_720p	HD 1280 x 720	1660	32.1	2:30	24

For the experimental setup, four test scenarios have been designed. Firstly, videos are sent using the best effort network without using the proposed VQoSRR. Secondly, the VQoSRR has been tested without using the traffic re-routing algorithm (DR-RA), only using the TwoLLWPs. The last two scenarios analyzed the effect of the DR-RA. In the first case, the video flow threshold was assigned to the first service group (A), which has hard QoS constraints and implies the flow must route over the Lowest Loss path. In the second case, the flow belonged to the second service group (B), which has soft constraints; in this situation, when no lowest loss path exists, the flow takes the Widest Path. The delivered video on each client is recorded and matched with the original video sequence using MSU Video Quality Measurement Tool.

Figure 5.12 illustrates a comparison of the four scenarios; it clarifies the effect on video quality in these cases. The results indicate that the SSIM average values of DR-RA provided high quality; the average SSIM of group A is around 0.97 for HD and about 0.96 for SD. The average SSIM values for group B are close to 0.94 for HD and 0.93 for SD. In comparison, when DR-RA was not applied, the video quality decreased, where the SSIM value of HD's average = 0.92 and SD's average = 0.93. Further, the best-effort results (without using the proposed VQoSRR) highly correlated with the degradation of video quality, where SSIM averages around 0.90 for HD and 0.91 for SD.

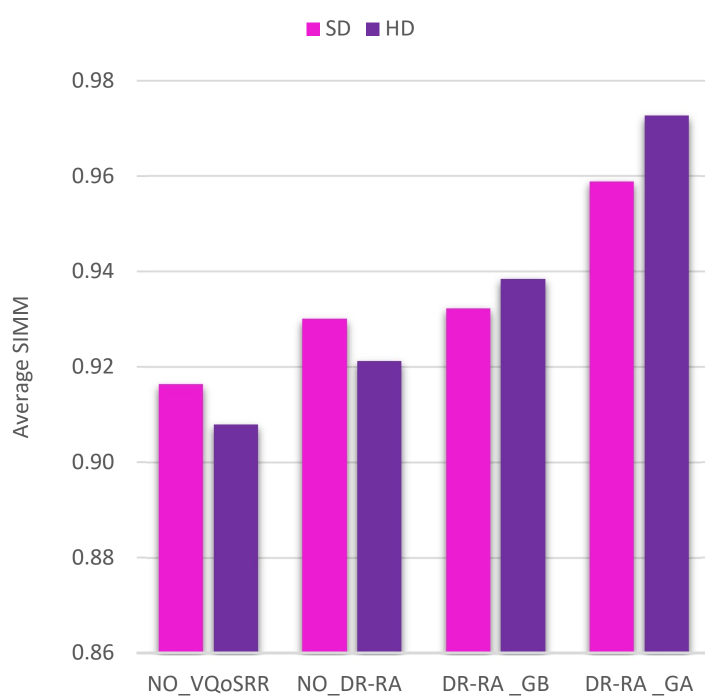

Figure 5.12: Average SSIM of the four testing scenarios

Figure 5.13 (a) and Figure 5.13 (b) present VMAF and SSIM measured during video delivery using Real-time Transport Protocol (RTP), with DR-RA, without DR-RA, and without the VQoSRR separately. A video delivered without DR-RA has higher VMAF and SSIM than a video delivered without VQoSRR. Additionally, when delivered using DR-RA, it has higher quality than both of them. The reason is that DR-RA can reroute video traffic based on network conditions by re-estimating packet loss probability and bandwidth utilization in order to comply with video QoS requirements, which in turn results in better video quality.

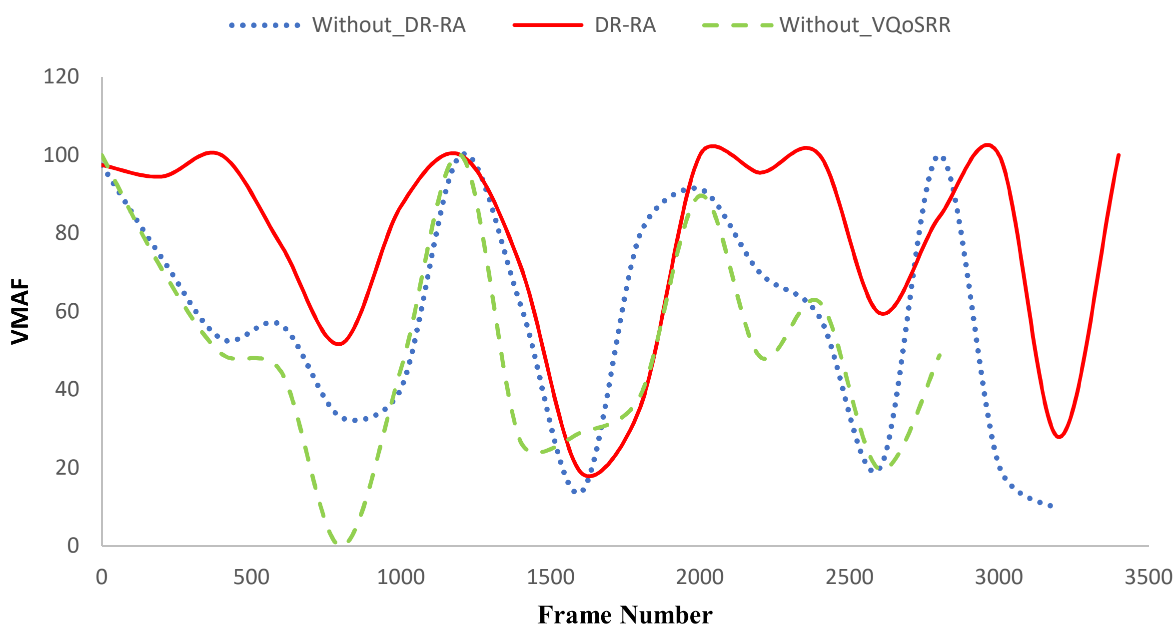

Figure 5.13 (a): Comparison of VMAF metrics when using DR-RA, without DR-RA, and without the proposed VQoSRR.

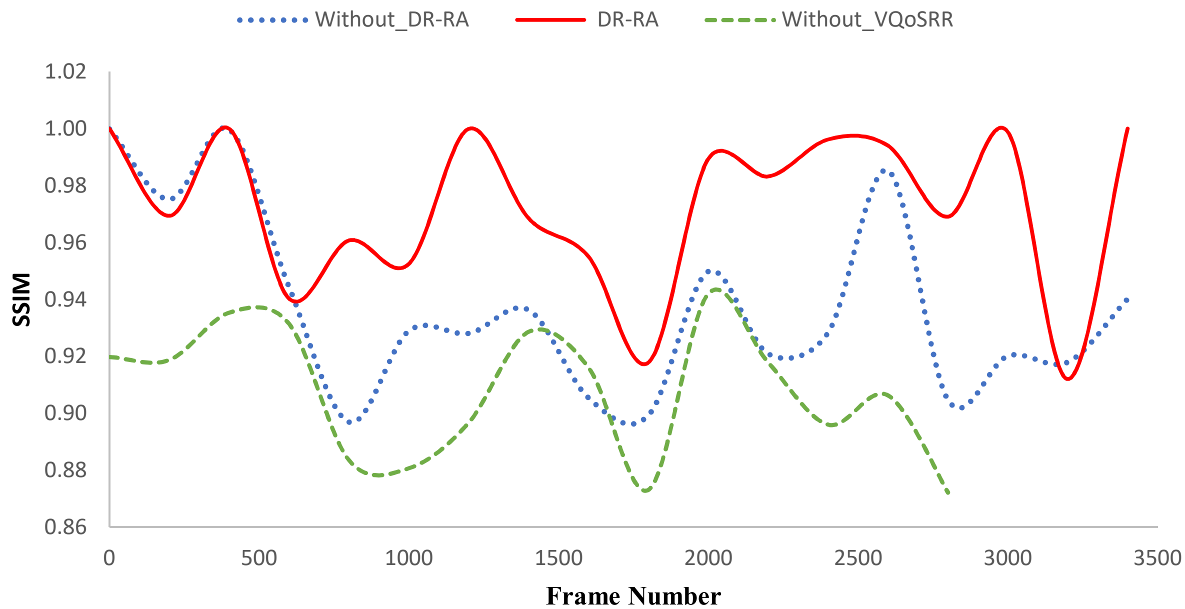

Figure 5.13 (b): Comparison of SSIM metrics when using DR-RA, without DR-RA, and without the proposed VQoSRR.

DR-RA is executed periodically at predetermined intervals. Hence, this section studies the effect of execution interval time on ongoing video traffic in order to analyze whether shifting traffic between one path and another affects video quality and which is the most optimal execution interval, as illustrated by Figures 5.14 and 5.15. In this study, interval times were (5, 7 or, 10 seconds). Figure 5.14 (a) and (b) present achieved results for the received SD video, respectively. The SSIM values are measured every 200 frames in Figure 5.14 (a) and every 10 frames in Figure 5.14 (b). It can be observed from the Figures that SSIM is higher when the interval (t = 5s) compared to (t = 7s) and (t = 10s); whereas there is no significant difference between t=7s and t=10s in terms of SSIM measurements.

Furthermore, Figure 5.15 (a) and Figure 5.15 (b) also show the effects of executing the dynamic rerouting algorithm at different time intervals on HD video quality measured by SSIM every 200 frames in Figure 5.15 (a) and every 10's frames in Figure 5.15 (b). As can be seen from the Figures, the DR-RA improves the SSIM when interval (t = 7s).

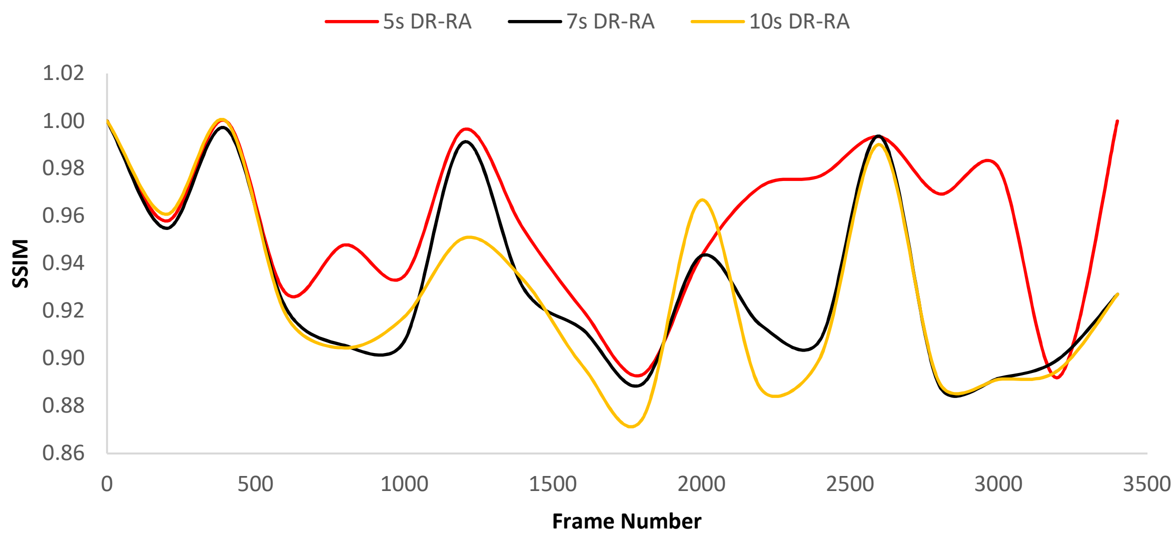

Figure 5.14 (a): Effects of executing dynamic rerouting algorithm at different times on the SD for every 200 frames.

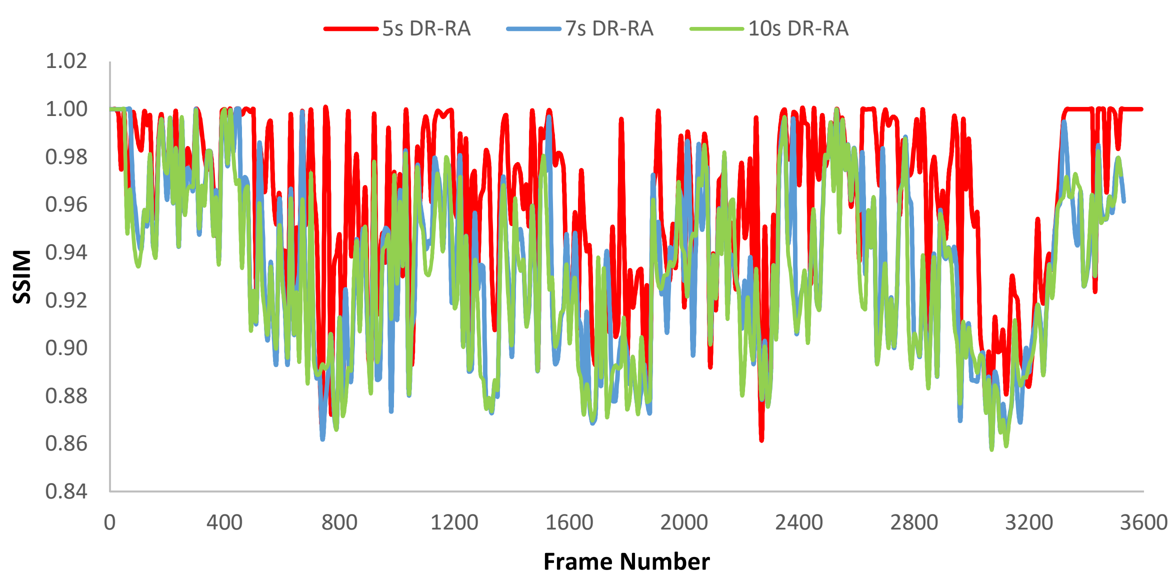

Figure 5.14 (b): Effects of executing dynamic rerouting algorithm at different times on the SD for every 10 frames

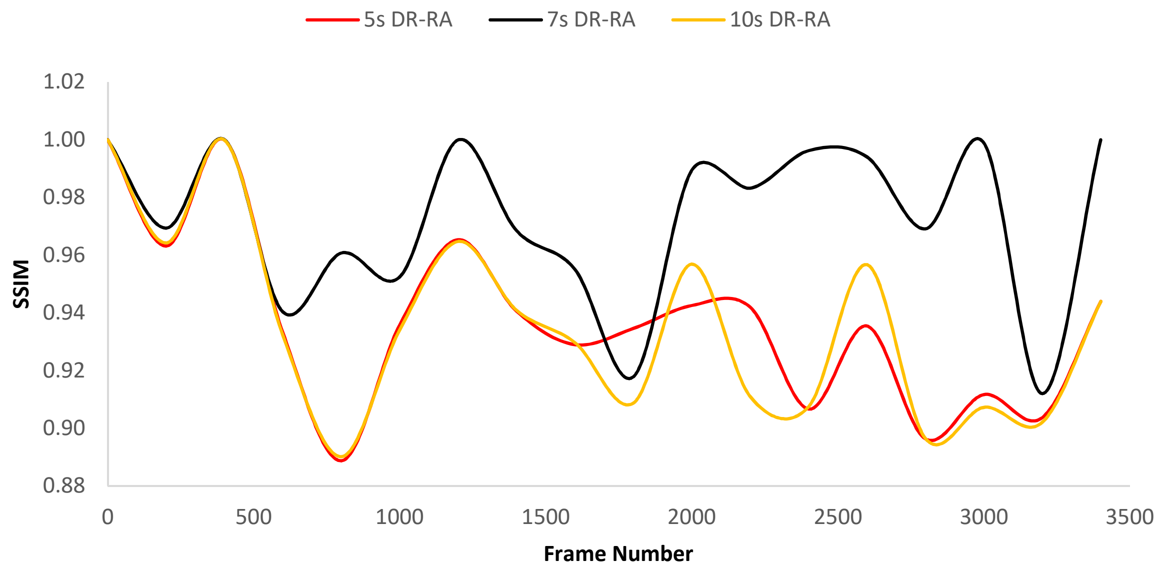

Figure 5.15 (a): Effects of executing dynamic rerouting algorithm at different times on the HD for every 200 frames.

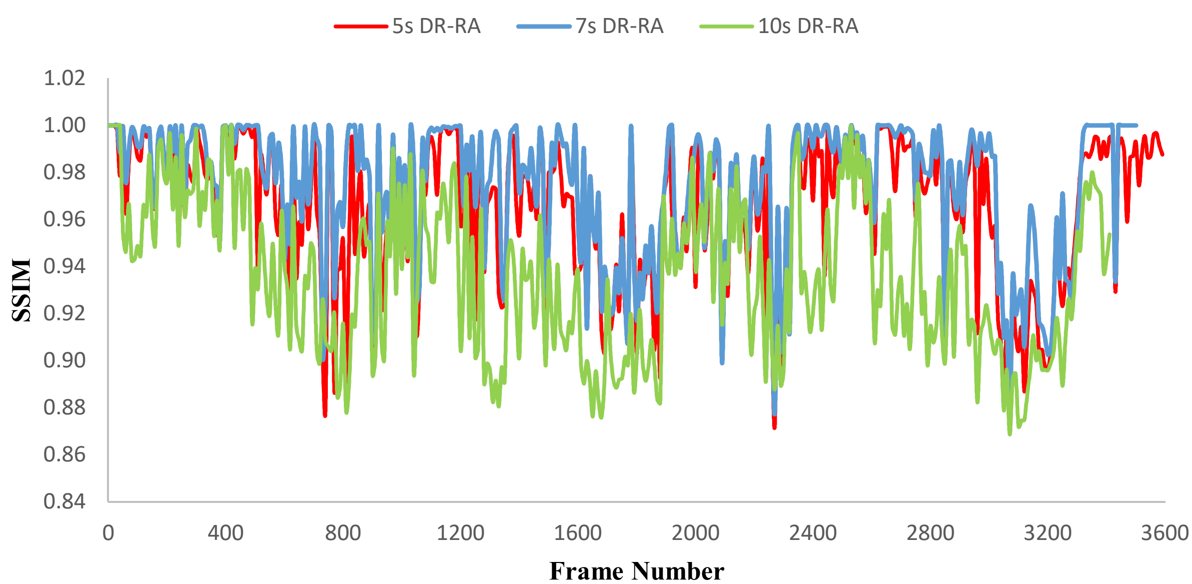

Figure 5.15 (b): Effects of executing dynamic rerouting algorithm at different times on the HD for every 10 frames.

The following conclusions have been reached from the above results. First, The HD achieves the highest SSIM at 7s, while the SD has the best result at 5s, which indicates that DR-RA should be executed at longer intervals for HD than SD in order to shift the flow to a new path.

Second, our experiment does not explore all possible video durations; long-duration videos may require further analysis to identify the most appropriate execution interval of the algorithm for the different video resolutions. Third, changing routes frequently can cause other QoS problems, such as increasing the jitter experienced by the end-users, so rerouting should not be subjected to frequent changes.

5.5 QoS-based routing with Reservation Experimental Results

To validate the proposed reservation methodology, an investigation has been performed to observe how the reservation is affecting video QoE. The following simulation parameters help to measure the performance of the suggested technique. This experiment used three video stream traffics in MP4 format; VQoSRR classified them into an HD video stream, SD video stream, and best-effort video stream. Both HD and SD have specific bandwidth and packet loss rate metrics. They are transmitted and decoded simultaneously in real-time through the network; their QoE parameters specifications have previously been described in Table 5.5. Additionally, on each OVS switch port connected to the network, three queues have been added. One queue is reserved for each of the VQoSRR flow types. The maximum capacity supported on every link is 50Mbps.

To obtain the results, we conducted three test cases. The first case used the controller default setting without considering QoS and traded all videos as best-effort (No_VQoSRR). The second case applied only the QoS-based routing proposed algorithm with the dynamic rerouting algorithm (VQoS-R). The third case combined the dynamic route computation with the per-class queue reservation scheme (VQoS-RR). Simulation parameters that are common for all the test cases are presented in Table 5.6. The proposed model was validated against the objective and subjective metrics. The video quality was measured by the objective metric (SSIM) and subjective (MOS) metrics in each test case. The MSU Video Quality Measurement Tool is used to measure the received video quality on each client against the original video sequence.

Table 5.6: Simulation Parameters

Video Type	No_VQoSRR	VQoS-R		VQoS-RR		
		Loss Metric	Bandwidth Metric	Loss Metric	Bandwidth Metric	Queues Rates
Video 1, SD	-	0.5%	3Mb/s	0.5%	3Mb/s	Estimated Around 30%
Video 2, HD	-	0.05%	10Mb/s	0.05%	10 Mb/s	Estimated Around 45%
Video 3, HD	-	Best Effort	Best Effort	Best Effort	Best Effort	Estimated Around 25%

Figure 5.16 (a) and Figure 5.16 (b) illustrate the SSIM calculated in the test cases for SD and HD, respectively. In the Figures, it can be seen that the proposed methods (VQoS-R and VQoS-RR) result in an improvement in the SSIM scores, and significantly, they are more efficient than the traditional controller shortest path method (No_VQoSRR); where the quality degradation is mostly observed. Besides, VQoS-RR offers a better quality video stream than VQoS-R.

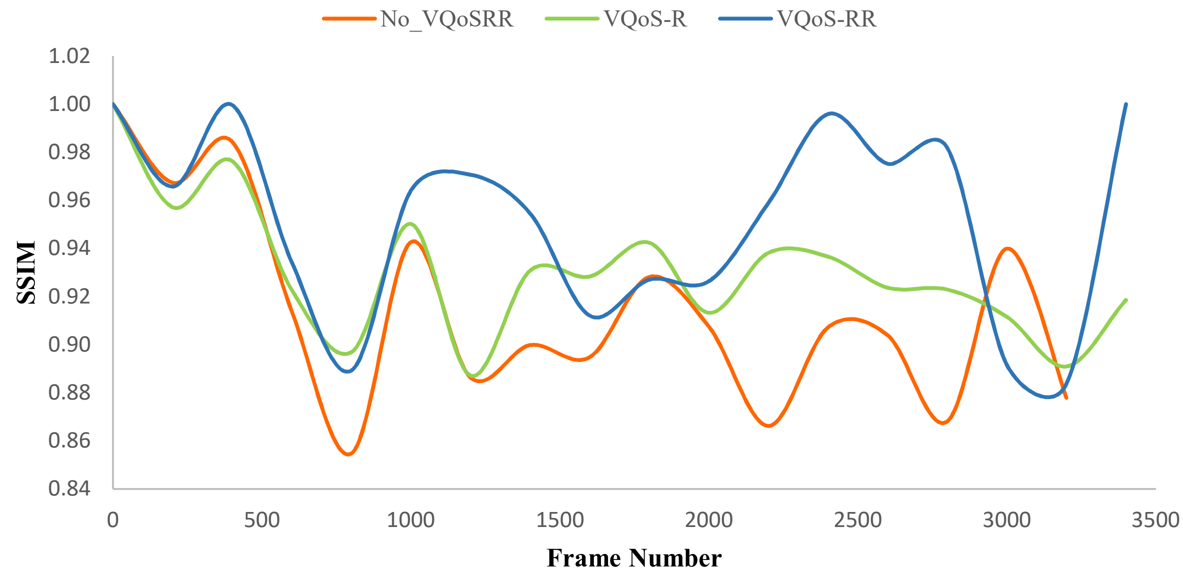

Figure 5.16 (a): Influence of test cases on SD video.

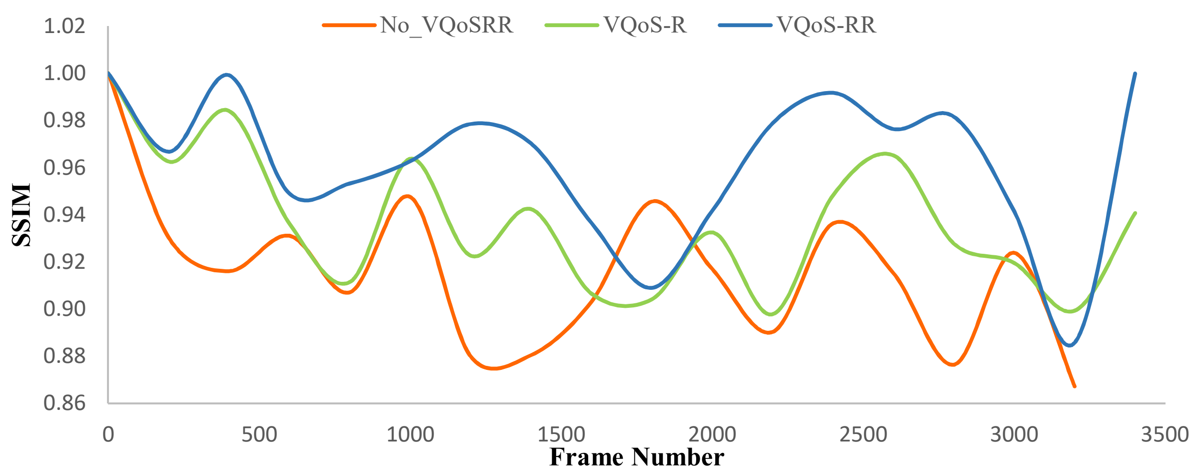

Figure 5.16 (b): Influence of test cases on the HD video.

Figure 5.17 shows the computed SSIM average values for the received video streams. As observed from the graph, VQoS-R and No_VQoSRR yielded lower SSIM scores compared to using VQoS-RR. As we have seen, the VQoS-RR provides good average values for all transmitted videos, including the best-effort videos. The reason is that the VQoS-RR method combines the reservation mechanism with QoS-based routing, which enhanced end-user perception of quality.

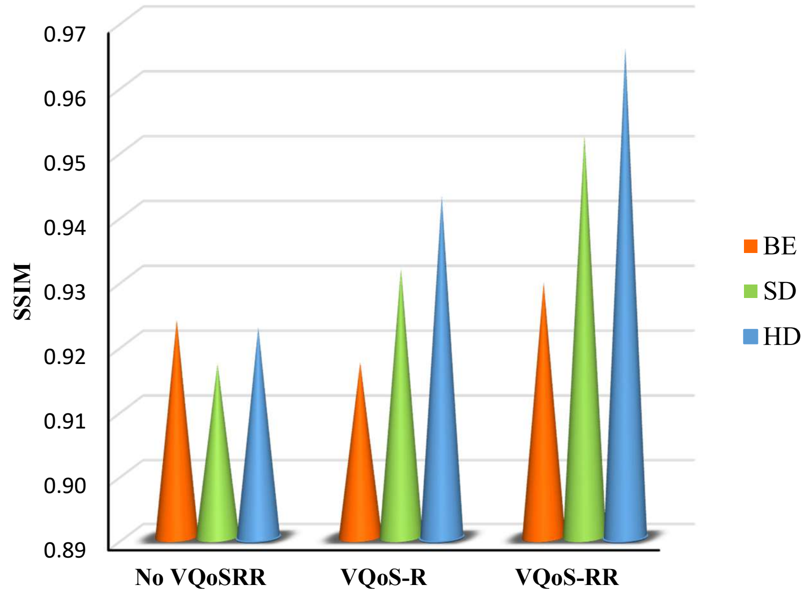

Figure 5.17: The mean SSIM value of the tested videos.

Below are the results obtained from the subjective QoE tests, where Figures 5.18 and 5.19 represent the MOS scores rated by users of 15 test conditions for each test case; these measurements are based on MOS scores (1-5).

Figure 5.18 (a) and Figure 5.18 (b) present a comparison of SD and HD videos with the MOS scores, respectively. As observed from the Figures, the distortion caused by the packet loss affected the measured MOS, where the values achieved without the VQoSRR were significantly lower compared to the VQoS-R and VQoS-RR. Take, for instance, Figure 5.18 (a), on average, MOS without the VQoSRR was about 40% of 5 compared to 61.3% for VQoS-R and 76% for VQoS-RR, respectively. Also, Figure 5.18 (b) presents a similar pattern for HD, where VQoS-RR scored 97.3%, and VQoS-R got 62.7, compared to No_VQoSRR, which achieved 41.3%.

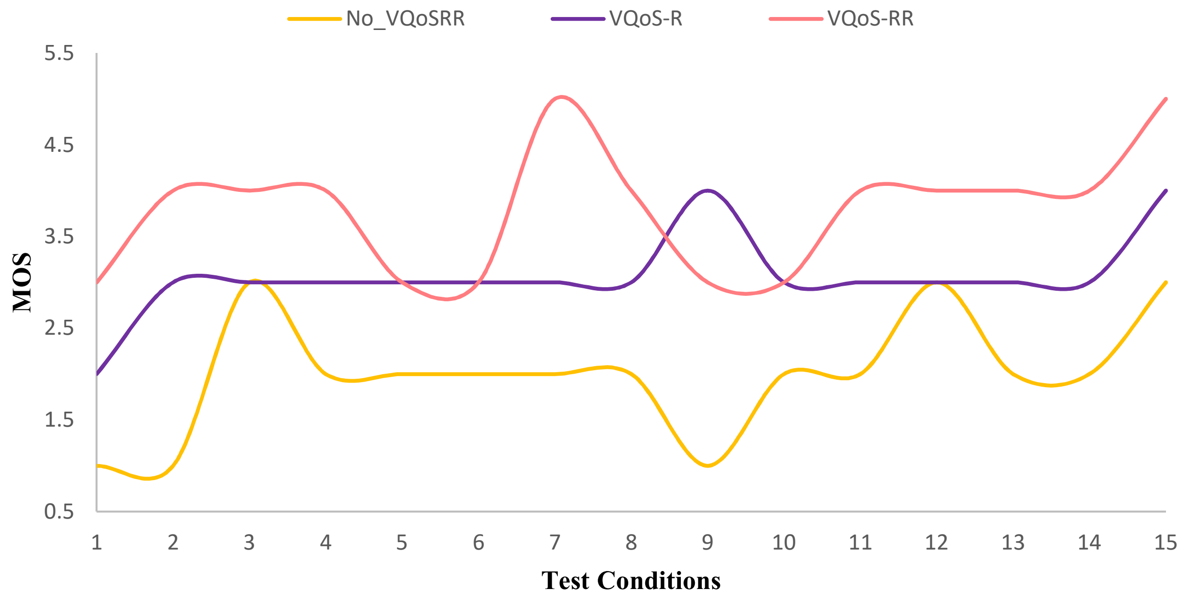

Figure 5.18 (a): The MOS values of the tested SD videos.

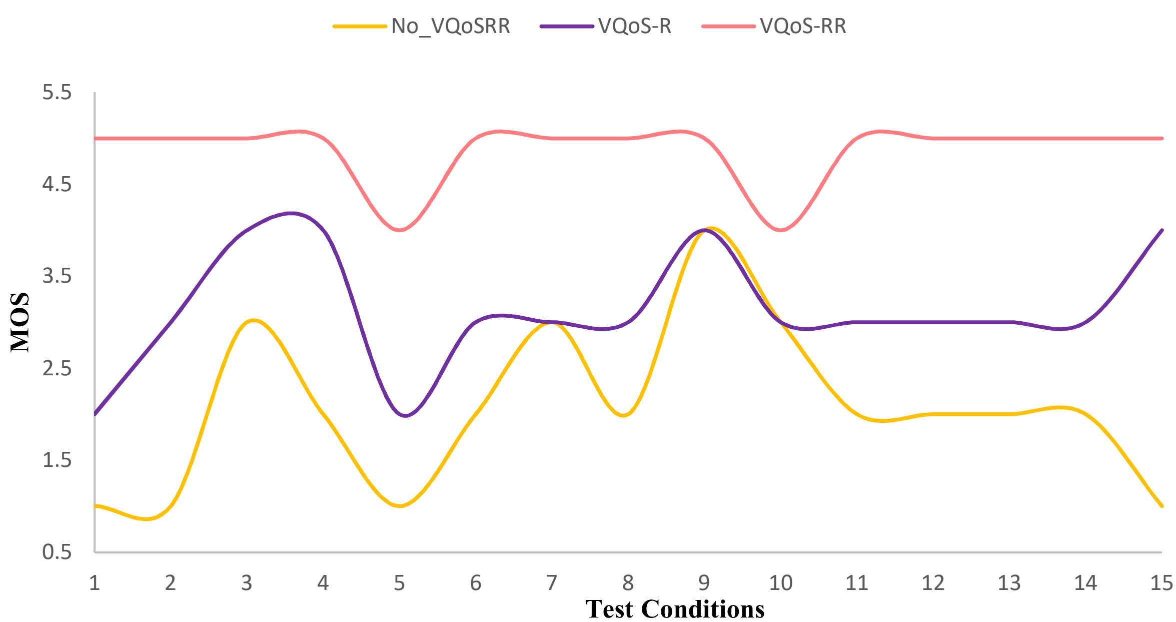

Figure 5.18 (b): The MOS values of the tested HD videos.

Figure 5.19 shows the MOS average values calculated to measure the quality difference between the three test cases. The MOS measured for No_VQoSRR reached 1.6 for best-effort video, and 2 for SD and 2.1 for HD, all of these values indicating (poor quality video). In terms of

VQoS-R, BE MOS was 1.8 (poor quality), SD MOS increased from 2 (poor quality) to 3 (fair quality), as well as HD improved from 2.1 (poor quality) to 3.1 (fair quality). While in VQoS-RR, the MOS remained at 1.6 (poor quality) for BE, 3.8 (fair to good quality) for SD, and 4.8 (good to excellent) for HD. As we can see from the Figure, the VQoS-RR improved the video quality; in HD, for example, the measured MOS increased from 41.3% to 97.3% over No_VQoSRR, which implies a 56% increase. Therefore, not using the VQoSRR will result in low MOS, which means severe video streaming quality degradation. Additionally, this result indicates that the proposed system succeeds in achieving good user perception.

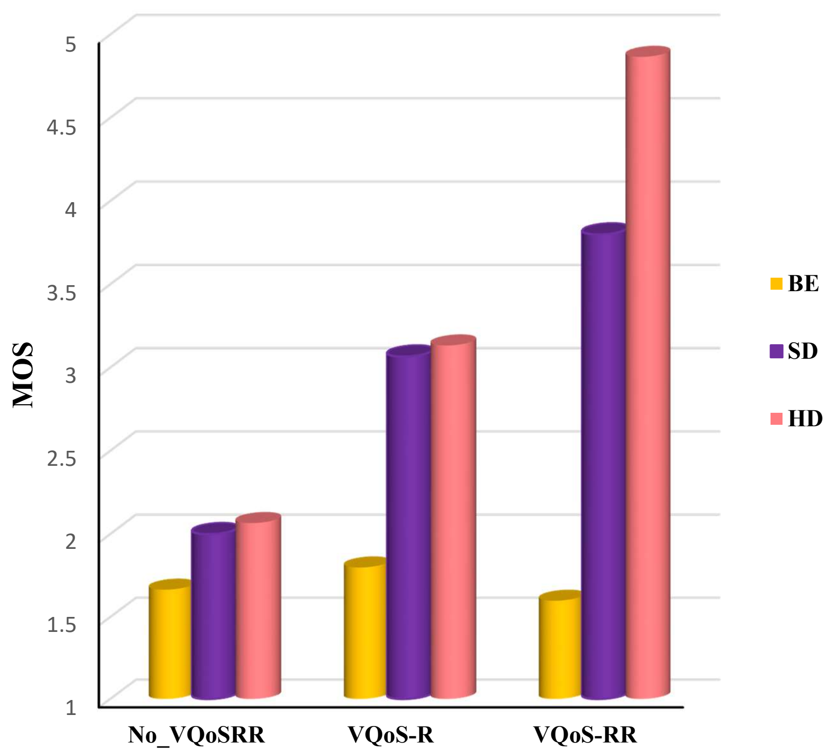

Figure 5.19: The average MOS value of the tested videos.

Figure 5.20 (a), Figure 5.20 (b), and Figure 5.20 (c) represent the correlation of the results obtained from the subjective tests against those obtained from the objective tests; for best-effort

(BE) video, SD video, and HD video, respectively. In this comparison, we mapped the values of objective SSIM metrics to an equivalent MOS scale (objective MOS).

In Figure 5.20 (a), the subjective MOS and objective MOS for BE traffic with and without VQoSRR are slightly correlated, with subjective MOS not exceeding 1.7 and objective MOS reaching around 3. Figure 5.20 (b) shows that SD in the case of VQoSRR has highly correlated MOS and SSIM results, whereas MOS without VQoSRR has a low quality compared to the objective MOS with fair quality. Finally, the results of HD video in Figure 5.20 (c) demonstrate that there is a poor approximation between SSIM and subjective MOS in both test cases. With VQoSRR, subjective MOS scored higher than objective MOS, while without it, the opposite happens. Further, when comparing this result of HD to SD in Figure 5.20 (a), we can see that subjective QoE tests had higher MOS ratings in HD, which reflects that most users prefer watching videos at high resolution.

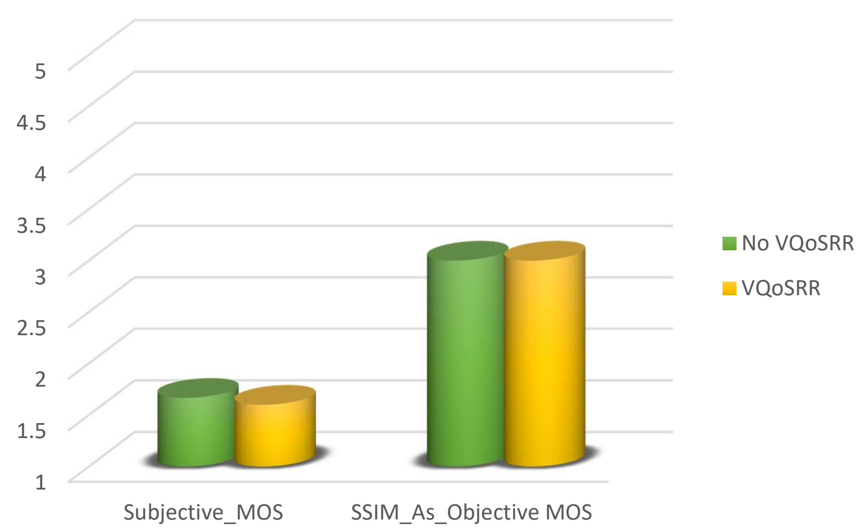

Figure 5.20 (a): Comparison between subjective and objective metrics for BE QoE.

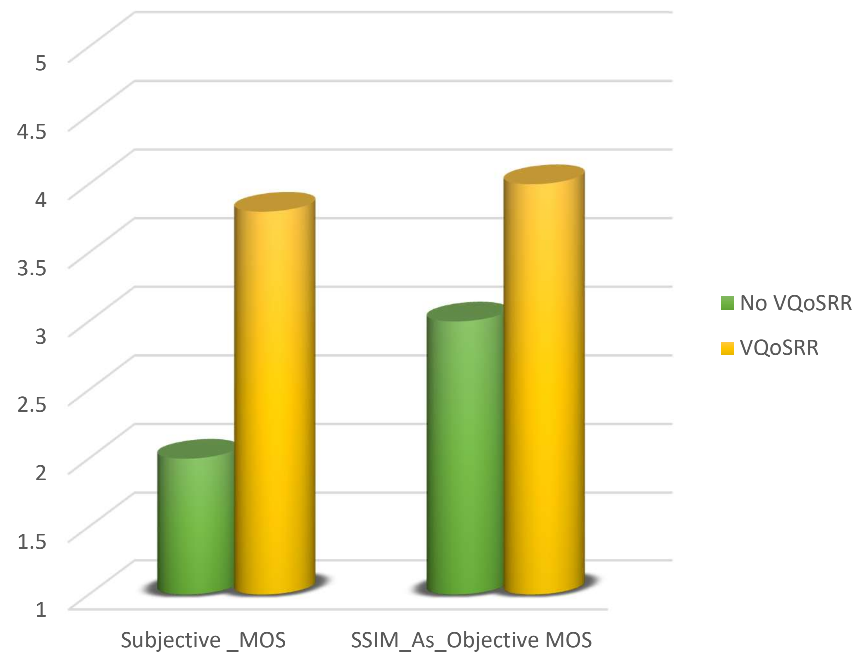

Figure 5.20 (b): Comparison between subjective and objective metrics for SD QoE.

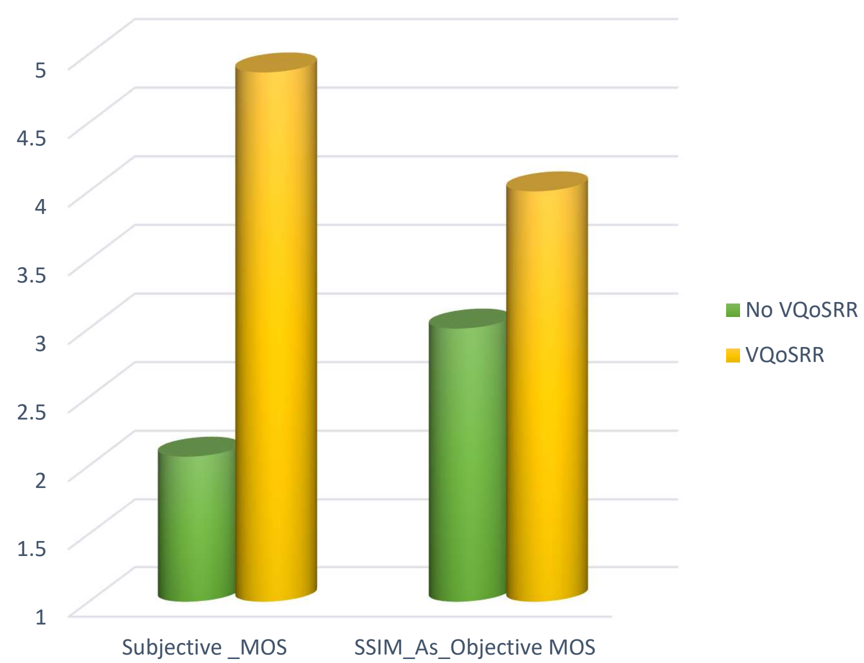

Figure 5.20 (c): Comparison between subjective and objective metrics for HD QoE.

In conclusion, an objective method shows lower video quality than subjective testing in HD, while the results were very close in the SD and the BE when the proposed model was applied. In contrast, BE, HD, and SD are not approximate without VQoSRR, where the subjective test indicates lower quality. We can notice that subjective MOS measurements reflect the differences in results more clearly than the objective SSIM measurements for comparing results obtained with and without the proposed model.

5.6 Results Comparison Summary

The video quality has been studied in experiments using both objective and subjective metrics. These experiments showed a better perception of video quality when using the VQoSRR method than the default settings for SDN controllers when delivering video based on resolution (SD and HD) QoS thresholds. These thresholds were packet loss and bandwidth. Furthermore, the following conclusions can be drawn: (1) generally, the number of video frames received by VQoSRR is higher than the number of frames received without using it, since the proposed QBR avoids routes that have the highest loss rate; (2) a lower received quality video results in degradation in the video viewing experience. For example, the perceived video quality achieved is highest in the case of VQoSRR for all transmitted videos, even the best effort videos. However, all of them experience more degradation in video viewing in the case of No_VQoSRR. (3) Tests of DR-RA cover only short video lengths; an analysis of long-duration videos may be necessary to determine the most appropriate interval of the algorithm. In addition, changing routes frequently by DR-RA can cause other QoS problems, such as increasing the delay variation experienced by end-users. Therefore, rerouting should not be subject to frequent changes. (4) The subjective MOS metric reflects the differences in results more clearly than the objective SSIM metric; (5) MOS test results indicate that VQoSRR improves video QoE by up to 97.3% and 76% for HD and SD video resolutions, respectively, compared to not using it. (6) We recommend evaluating the effectiveness of VQoSRR's QoS scalability when there is a large amount of high-priority videos.

CHAPTER VI

CONCLUSION AND FUTURE WORK

6.1 Conclusion

The VQoSRR is a QoS and QoE-aware SDN framework for video streaming that adjusts network resources and service characteristics based on the user's needs. The proposed method combines the reservation mechanism with QoS-based routing. This allows the determining of which network path has adequate resources to support the requested QoS, as well as providing reserving and requesting network resources at the same time.

The main goal of the routing strategy was to satisfy two routing constraints: Packet loss and Bandwidth. It efficiently determines feasible paths for each video flow according to the current state of the entire network. We took advantage of SDN to implement our state-dependent routing scheme by benefiting from the controller's global view of the network. This study focuses on investigating packet loss and bandwidth metrics associated with HD and SD. Considering that the routing algorithm depends on thresholds defined by the administrator, all other QoS requirements related to streaming videos, including 4K video resolution, can be implemented if the policy defines their bandwidth and loss metrics.

We used SSIM, VMAF, and MOS QoE measurements for SD and HD videos to evaluate the proposed framework's performance. The experiments proved that VQoSRR enhanced the user perception of quality and allowed fine control over routes and resources. Moreover, the results suggest that HD videos are impacted more than SD videos when there is more packet loss because it obtains more frame artifacts and color distortions.

From the results, the video QoE was improved. For instance, for SSIM measurements, when we used VQoSRR, the SSIM quality value increased from 0.91 to 0.95 for SD videos, whereas for HD videos, it increased from 0.92 to 0.96. Likewise, in the case of MOS measurements, the VQoSRR increased the average MOS by 36.0% for SD and 56% for HD. We also observed that subjective MOS reflected the differences in results more clearly than the objective SSIM for comparing results obtained with the proposed model.

6.2 Future Work

There are still several issues related to this area of research that need to be explored, as well as several ways in which this thesis could be improved and extended. These issues and suggestions can be addressed as follows:

- Evaluating the performance of the proposed scheme in real network environment.
- Our experiment did not explore all possible video durations; therefore, long-duration videos may require further analyses to identify the most appropriate execution interval of the algorithm for the different video resolutions.
- More work is needed to investigate the scalability issues in a large-scale network environment such as deploying the proposed model across inter-domains.
- Integrating the proposed model with multicast and broadcast transmission.

Bibliography

Al Hasrouty, C., 2018. Adaptive Multicast Live Streaming for A/V Conferencing Systems over Software-Defined Networks (Doctoral dissertation, Université de Bordeaux).

Al-Madani, B., Al-Roubaiey, A. and Baig, Z.A., 2014. Real-time QoS-aware video streaming: a comparative and experimental study. Advances in Multimedia, 2014.

Alreshoodi, M.A., 2016. Prediction of Quality of Experience for Video Streaming Using Raw QoS Parameters (Doctoral dissertation, University of Essex).

Arreaza Govea, R., 2019. Rate-controlled Low Latency Service with OpenFlow.

Babiarz, J., Chan, K. and Baker, F., 2006. Configuration guidelines for DiffServ service classes. Network Working Group.

Blake, S., Black, D., Carlson, M., Davies, E., Wang, Z., Weiss, W., December, 1998. An architecture for differentiated services. RFC 2475. www.rfc-archieve.org.

Bouillet, E., Ellinas, G., Labourdette, J.F. and Ramamurthy, R., 2007. Path routing in mesh optical networks. Hoboken, NJ: John Wiley & Sons.

Braden, R., Clark, D., Shenker, S., June, 1994. RFC1633: integrated services in the Internet architecture: an overview. Request for Comments (RFC) 1633 (IETF). www. rfc-archieve.org.

Caminandes, 2021 [Online]. Available at: http://www.caminandes.com/.

Canovas, A., Rego, A., Romero, O. and Lloret, J., 2020. A robust multimedia traffic SDN-Based management system using patterns and models of QoE estimation with BRNN. Journal of Network and Computer Applications, 150, p.102498.

Chen, Y., Wu, K. and Zhang, Q., 2014. From QoS to QoE: A tutorial on video quality assessment. IEEE Communications Surveys & Tutorials, 17(2), pp.1126-1165.

Cisco (2016)., "Visual Network Index (VNI) Complete Forecast Highlights," 2021; https://www.cisco.com/c/dam/m/en_us/solutions/service-provider/vni-forecast-highlights/pdf/Global_2021_Forecast_Highlights.pdf, p. 3, (Accessed: 12 October 2021).

Cisco 2013., "Cisco Video Surveillance Solution Reference Network Design Guide" June. https://www.cisco.com/c/dam/en/us/td/docs/security/physical_security/video_surveillance/network/vsm/7_0/srnd/vsm_7_0_srnd.pdf.

Civanlar, S., Parlakisik, M., Tekalp, A.M., Gorkemli, B., Kaytaz, B. and Onem, E., 2010, December. A qos-enabled openflow environment for scalable video streaming,IEEE Globecom Workshops on Network of the Future (FutureNet-III) (pp. 351-356), Miami.. IEEE.

Costa, L.H.M.K., Fdida, S., Duarte, O.C.M.B., 2001. June. A scalable algorithm for link-state QoS-based routing with three metrics. In: ICC 2001. IEEE International Conference on Communications. Conference Record (Cat. No. 01CH37240), vol. 8. IEEE, pp. 2603–2607.

Costa, L.K., Duarte, O.B., 1999. June. A scalable QoS-based routing mechanism for supporting multimedia applications. Proceed. IEEE Int. Conf. Multimedia Comput. Syst. ICMSC'99 2, 346–351.

Crawley, E., Nair, R., Rajagopalan, B., Sandick, H., 1998. A framework for QoS-based routing in the Internet. RFC 2386. www.rfc-archieve.org.

Dijkstra, E.W., 1959. A note on two problems in connexion with graphs. Numerische mathematik, 1(1), pp.269-271.

Egilmez, H.E., Civanlar, S., Tekalp, A.M., April, 2013. An optimization framework for QoS-enabled adaptive video streaming over OpenFlow networks. IEEE Trans. Multimed. 15 (3), 710–715.

Egilmez, H.E., Dane, S.T., Bagci, K.T., Tekalp, A.M., 2012. December. OpenQoS: an OpenFlow controller design for multimedia delivery with end-to-end quality of service over software-defined networks. In: Proceedings of the 2012 Asia Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC). IEEE, pp. 1–8.

Egilmez, H.E., Gorkemli, B., Tekalp, A.M., Civanlar, S., 2011. September. Scalable video streaming over OpenFlow networks: an optimization framework for QoS routing. In: 2011 18th IEEE International Conference on Image Processing. IEEE, Brussels, Belgium, pp. 2241–2244.

Feamster, N., Rexford, J. and Zegura, E., 2014. The road to SDN: an intellectual history of programmable networks. ACM SIGCOMM Computer Communication Review, 44(2), pp.87-98.

Floodlight. 2020. Available online: https://floodlight.atlassian.net/wiki/spaces/floodlightcontroller/overview (accessed on 8 June 2021).

Ghalwash, H. and Huang, C.H., 2018, October. A QoS framework for SDN-based networks. In 2018 IEEE 4th International Conference on Collaboration and Internet Computing

(CIC) (pp. 98-105). IEEE.
Go, S.J.Y., Festin, C.A.M., Tan, W.M., 2019. An SDN-based framework for improving the performance of underprovisioned IP Video Surveillance networks. J. Netw. Comput. Appl. 132, 49–74.
Gries, V., 1996. Analysing the quantization parameters for quality assessment of digital video. In Multimedia'96 (pp. 53-67). Springer, Vienna.
Gstreamer. 2020. Available online: https://gstreamer.freedesktop.org/ (accessed on 15 June 2020).
Guo, Y., Wang, W., Zhang, H., Guo, W., Wang, Z., Tian, Y., Yin, X., Wu, J., 2021. Traffic engineering in hybrid software defined network via reinforcement learning. J. Netw. Comput. Appl. 189, 103116.
Hands, D., Bourret, A. and Bayart, D., 2005. Video QoS enhancement using perceptual quality metrics. BT Technology Journal, 23(2), pp.208-216.
Hekstra, A.P., Beerends, J.G., Ledermann, D., De Caluwe, F.E., Kohler, S., Koenen, R.H., Rihs, S., Ehrsam, M. and Schlauss, D., 2002. PVQM–A perceptual video quality measure. Signal processing: Image communication, 17(10), pp.781-798.
Henni, D.E., Ghomari, A. and Hadjadj-Aoul, Y., 2020. A consistent QoS routing strategy for video streaming services in SDN networks. International Journal of Communication Systems, 33(10), p.e4177.
Huynh-Thu, Q. and Ghanbari, M., 2012. The accuracy of PSNR in predicting video quality for different video scenes and frame rates. Telecommunication Systems, Vol. 49, No. 1, pp.35.
Iperf, TCP/UDP Bandwidth Measurement Tool, 2021 [Online]. Available at: https://iperf.fr/.
ITU, I., 2008. 1080, Quality of Experience Requirements for IPTV Services https://www.itu.int/rec/T-REC-G.1080-200812-I.
ITU-T, R.Y., 2011. 1541,". Network performance objectives for IP-based services.
Jammal, M., Singh, T., Shami, A., Asal, R. and Li, Y., 2014. Software defined networking: State of the art and research challenges. Computer Networks, 72, pp.74-98.
Jonnalagadda, V. and Musti, V., 2012. Evaluation of video quality of experience using evalvid. Electrical Engineering.
Kaur, G., Kaur, N. and Tiwana, P.S.,2016. Survey of UDP, TCP for video streaming in WiMAX.
Khan, A., Sun, L. and Ifeachor, E., 2008, September. An ANFIS-based hybrid video quality

prediction model for video streaming over wireless networks. In 2008 The Second International Conference on Next Generation Mobile Applications, Services, and Technologies (pp. 357-362). IEEE.

Khater, A. and Hashemi, M.R., 2018, May. Dynamic Flow Management Based on DiffServ in SDN Networks. In Electrical Engineering (ICEE), Iranian Conference on (pp. 1505-1510). IEEE.

Kim, H.J. and Choi, S.G., 2010, February. A study on a QoS/QoE correlation model for QoE evaluation on IPTV service. In 2010 The 12th International Conference on Advanced Communication Technology (ICACT) (Vol. 2, pp. 1377-1382). IEEE.

Kreutz, D., Ramos, F.M., Esteves Verissimo, P., Esteve Rothenberg, C., Azodolmolky, S. and Uhlig, S., 2015. "Software-defined networking: A comprehensive survey". proceedings of the IEEE, 103(1), pp.14-76.

Lantz, B., Heller, B. and McKeown, N., 2010, October. A network in a laptop: rapid prototyping for software-defined networks. In Proceedings of the 9th ACM SIGCOMM Workshop on Hot Topics in Networks (pp. 1-6).

Li, Z., Aaron, A., Katsavounidis, I., Moorthy, A. and Manohara, M., 2016. Toward a practical perceptual video quality metric. The Netflix Tech Blog, 6(2).

Li, Z., Bampis, C., Novak, J., Aaron, A., Swanson, K., Moorthy, A. and Cock, J.D., 2018. VMAF: The journey continues. Netflix Technology Blog.

Liu, W.X., Cai, J., Chen, Q.C., Wang, Y., 2021. DRL-R: deep reinforcement learning approach for intelligent routing in software-defined data-center networks. J. Netw. Comput. Appl. 177, 102865.

Lloret, J., Garcia, M., Atenas, M. and Canovas, A., 2011. A QoE management system to improve the IPTV network. International Journal of Communication Systems, 24(1), pp.118-138.

Mallesh, S., 2017. Automatic detection of elephant flows through openflow-based openvswitch (Doctoral dissertation, Dublin, National College of Ireland).

Masip-Bruin, X., Yannuzzi, M., Domingo-Pascual, J., Fonte, A., Curado, M., Monteiro, E., Kuipers, F., Van Mieghem, P., Avallone, S., Ventre, G. and Aranda-Gutiérrez, P., 2006. Research challenges in QoS routing. Computer communications, 29(5), pp.563-581.

McCabe, J.D., 2010. Network Analysis, Architecture, and Design. Morgan Kaufmann.

McKeown, N., Anderson, T., Balakrishnan, H., Parulkar, G., Peterson, L., Rexford, J., Shenker,

S. and Turner, J., 2008. OpenFlow: enabling innovation in campus networks. ACM SIGCOMM Computer Communication Review,38(2), pp.69-74.

McKeown, N., Anderson, T., Balakrishnan, H., Parulkar, G., Peterson, L., Rexford, J., Shenker, S. and Turner, J., 2008. OpenFlow: enabling innovation in campus networks. ACM SIGCOMM Computer Communication Review,38(2), pp.69-74.

Mohanta, K. and Poddar, B., 2012. Comprehensive study on computational methods for k-shortest paths problem. International Journal of Computer Applications, 40(14), pp.22-26.

Mok, R.K., Chan, E.W. and Chang, R.K., 2011, May. Measuring the quality of experience of HTTP video streaming. In 12th IFIP/IEEE International Symposium on Integrated Network Management (IM 2011) and Workshops (pp. 485-492). IEEE.

MSU Video Quality Measurement Tool, 2021 [Online]. Available at: https://www.compression.ru/video.

Nam, H., Kim, K.H., Kim, J.Y. and Schulzrinne, H., 2014, December. Towards QoE-aware video streaming using SDN. In 2014 IEEE global communications conference (pp. 1317-1322). IEEE.

Netflix, 2020 [Online]. Available at: https://help.netflix.com/en/node/306?q=super+hd&ba=GSButtonClick, and https://www.sony.co.th/en/electronics/support/televisions-projectors-lcd-tvs/klv-21sr2/articles/00022091.

Nunes, M. Mendonca, X.-N. Nguyen, K. Obraczka and T. Turletti , "A survey of software-defined networking: Past, present, future of programmable networks" , IEEE Commun. Surveys Tuts., vol. 16 , no. 3 , pp.1617 -1634 , 2014.

ONF, 2014. “SDN architecture”, Issue 1 June, 2014 ONF TR-502.

Open vSwitch, 2020 [Online]. Available at: https://www.openvswitch.org/

Owens II, H. and Durresi, A., 2015. Video over software-defined networking (vsdn). Computer Networks, 92, pp.341-356. [CrossRef]

P. ITU-T RECOMMENDATION, 2008. Subjective video quality assessment methods for multimedia applications. Available online: https://www.itu.int/rec/T-REC-P.910 (accessed on 15 July 2021).

Parsaei, M.R., Boveiri, H.R., Javidan, R. and Khayami, R., 2020. Telesurgery QoS improvement over SDN based on a Type-2 fuzzy system and enhanced cuckoo optimization algorithm.

International Journal of Communication Systems, 33(11), p.e4426.

Pokhrel, J. (2015). Intelligent quality of experience (QoE) analysis of network served multimedia and web contents. In Institut National desTélécommunications.

Ranjan, P., Pande, P., Oswal, R., Qurani, Z. and Bedi, R., 2014. “A Survey of Past, Present and Future of Software Defined Networking”. International Journal, 2(4).

Recommendation, I.T.U.T., 2008. E. 800, Definitions of terms related to the quality of service. International Telecommunication Union’s Telecommunication Standardization Sector (ITU-T) Std.

Recommendation, I.T.U.T.P., 2008. 10/G. 100 Amendment 2: New definitions for inclusion in Recommendation ITU-TP. 10/G. 100. Vocabulary for performance and quality of service. Int. Telecomm. Union, Geneva.

Recommendation, I.T.U.T.P., 2017. 1203.3, “Parametric bitstream-based quality assessment of progressive download and adaptive audiovisual streaming services over reliable Transport-Quality integration module,”. International Telecommunication Union: Geneva, Switzerland.

Rego, A., Garcia, L., Sendra, S. and Lloret, J., 2018. Software Defined Network-based control system for an efficient traffic management for emergency situations in smart cities. Future Generation Computer Systems, 88, pp.243-253.

Rego, A., Sendra, S., Jimenez, J.M. and Lloret, J., 2019. Dynamic metric OSPF-based routing protocol for software defined networks. Cluster Computing, 22(3), pp.705-720.

Rego, A., Sendra, S., Jimenez, J.M., Lloret, J., 2017. May. OSPF routing protocol performance in software defined networks. In: 2017 Fourth International Conference on Software Defined Systems (SDS). IEEE, Valencia, Spain, pp. 131–136.

Rosen, E., Rekhter, Y., March, 1999. RFC2547: BGP/MPLS VPNs. RFC 2547. www.rfc-archieve.org.

Schulzrinne, H., Casner, S., Frederick, R. and Jacobson, V., 2003. RTP: A transport protocol for real-time applications (No. rfc3550).

Schulzrinne, H., Rao, A. and Lanphier, R., 1998. RFC2326: Real time streaming protocol (RTSP).

Seddiki, M.S., Shahbaz, M., Donovan, S., Grover, S., Park, M., Feamster, N. and Song, Y.Q., 2014, August. FlowQoS: QoS for the rest of us. In Proceedings of the third workshop on Hot topics in software defined networking (pp. 207-208).

Seitz, N., 2003. ITU-T QoS standards for IP-based networks. IEEE Communications Magazine, 41(6), pp.82-89.

Sendra, S., Rego, A., Lloret, J., Jimenez, J.M., Romero, O., 2017. May. Including artificial intelligence in a routing protocol using software defined networks. In: 2017 IEEE International Conference on Communications Workshops (ICC Workshops). IEEE, Paris, France, pp. 670–674.

Sezer, S., Scott-Hayward, S., Chouhan, P.K., Fraser, B., Lake, D., Finnegan, J., Viljoen, N., Miller, M. and Rao, N., 2013. Are we ready for SDN? Implementation challenges for software-defined networks. IEEE Communications Magazine, 51(7), pp.36-4.

Sharma, S., Staessens, D., Colle, D., Palma, D., Goncalves, J., Figueiredo, R., Morris, D., Pickavet, M. and Demeester, P., 2014, September. Implementing quality of service for the software defined networking enabled future internet. In 2014 third European workshop on software defined networks (pp. 49-54). IEEE.

Shi, Z., Zou, H., Rank, M., Chen, L., Hirche, S. and Muller, H.J., 2009. Effects of packet loss and latency on the temporal discrimination of visual-haptic events. IEEE Transactions on Haptics, 3(1), pp.28-36.

Sintel Trailer, 2021 [Online]. Available at: https://durian.blender.org/.

SmartDraw, 2020 [Online]. Available at: https://www.smartdraw.com/about/.

Stallings, W., 2013. “Software-defined networks and OpenFlow”. The Internet Protocol Journal, 16(1), pp.2-14.

Streijl, R.C., Winkler, S. and Hands, D.S., 2016. Mean opinion score (MOS) revisited: methods and applications, limitations and alternatives. Multimedia Systems, 22(2), pp.213-227.

Sullivan, G.J., Ohm, J.R., Han, W.J. and Wiegand, T., 2012. Overview of the high efficiency video coding (HEVC) standard. IEEE Transactions on circuits and systems for video technology, 22(12), pp.1649-1668.

Sullivan, G.J., Topiwala, P.N. and Luthra, A., 2004. The H. 264/AVC advanced video coding standard: Overview and introduction to the fidelity range extensions. Applications of Digital Image Processing XXVII, 5558, pp.454-474.

Tan, X., Guo, Y., Orgun, M.A., Xue, L. and Chen, Y., 2018. An engagement model based on user interest and QoS in video streaming systems. Wireless Communications and Mobile Computing, 2018.

Van den Branden Lambrecht, C.J. and Verscheure, O., 1996, March. Perceptual quality measure using a spatiotemporal model of the human visual system. In Digital Video Compression: Algorithms and Technologies 1996 (Vol. 2668, pp. 450-461). SPIE.

Van der Merwe, J., Sen, S. and Kalmanek, C., 2002, August. Streaming video traffic: Characterization and network impact. In Proceedings of the Seventh International Web Content Caching and Distribution Workshop.

Vega, M.T., Perra, C. and Liotta, A., 2018. Resilience of video streaming services to network impairments. IEEE Transactions on Broadcasting, 64(2), pp.220-234.

VLC media player. 2021. Available online: https://www.videolan.org/vlc/ (accessed on 2 February 2021).

Volpato, F., Da Silva, M.P., Gonçalves, A.L. and Dantas, M.A.R., 2017, July. An autonomic QoS management architecture for software-defined networking environments. In 2017 IEEE Symposium on Computers and Communications (ISCC) (pp. 418-423). IEEE.

Wang, Z. and Crowcroft, J., 1996. Quality-of-service routing for supporting multimedia applications. IEEE Journal on selected areas in communications, 14(7), pp.1228-1234.

Wang, Z., Lu, L. and Bovik, A.C., 2004. Video quality assessment based on structural distortion measurement. Signal processing: Image communication, 19(2), pp.121-132.

Winkler, S. and Mohandas, P., 2008. The evolution of video quality measurement: From PSNR to hybrid metrics. IEEE transactions on Broadcasting, 54(3), pp.660-668.

Xu, C., Chen, B., Qian, H., 2015. Quality of service guaranteed resource management dynamically in software defined network. J. Commun. 10 (11), 843–850. https://doi.org/10.12720/jcm.10.11.843-850.

Yamansavascilar, B., Baktir, A.C., Ozgovde, A., Ersoy, C., November, 2020. Fault tolerance in SDN data plane considering network and application based metrics. J. Netw. Comput. Appl. 170, 102780.

Yan, J., Zhang, H., Shuai, Q., Liu, B., Guo, X., 2015. HiQoS: an SDN-based multipath QoS solution. China Commun. 12 (5), 123–133.

Yen, J.Y., 1970. An algorithm for finding shortest routes from all source nodes to a given destination in general networks. Quarterly of Applied Mathematics, 27(4), pp.526-530.

Yu, T.F., Wang, K., Hsu, Y.H., 2015. January. Adaptive routing for video streaming with QoS support over SDN networks. In: 2015 International Conference on Information Networking

(ICOIN). IEEE, Cambodia, pp. 318–323. https://doi.org/10.1109/ICOIN.2015.7057904.

Yu, Y.S. and Ke, C.H., 2018. Genetic algorithm-based routing method for enhanced video delivery over software defined networks. International Journal of Communication Systems, 31(1), p.e3391.

Zinner, T., Hohlfeld, O., Abboud, O. and Hoßfeld, T., 2010, June. Impact of frame rate and resolution on objective QoE metrics. In 2010 second international workshop on quality of multimedia experience (QoMEX) (pp. 29-34). IEEE.